

**IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
MARSHALL DIVISION**

VALTRUS INNOVATIONS LTD. and
KEY PATENT INNOVATIONS LTD.,

Plaintiffs,

v.

LENOVO GROUP LIMITED, LENOVO
(SHANGHAI) ELECTRONICS
TECHNOLOGY CO. LTD., LENOVO
INFORMATION PRODUCTS
(SHENZHEN) CO. LTD., LCFC (HEFEI)
ELECTRONICS TECHNOLOGY CO. LTD.
D/B/A LC FUTURE CENTER AND
LENOVO COMPAL FUTURE CENTER,
LENOVO CENTRO TECNOLÓGICO S.
DE R.L. DE C.V., LENOVO PC HK LTD.,
LENOVO (BEIJING) LIMITED, LENOVO
(THAILAND) LIMITED, LENOVO
ENTERPRISE TECHNOLOGY COMPANY
LIMITED, LENOVO NETAPP
TECHNOLOGY LIMITED, MOTOROLA
MOBILE COMMUNICATION
TECHNOLOGY LTD., and MOTOROLA
(WUHAN) MOBILITY TECHNOLOGIES
COMMUNICATION COMPANY
LIMITED,

Defendants.

Case No. 2:25-cv-

JURY TRIAL DEMANDED

COMPLAINT FOR PATENT INFRINGEMENT

Plaintiffs Valtrus Innovations Ltd. (“Valtrus”) and Key Patent Innovations Ltd. (“KPI”) (collectively, “Plaintiffs”) for their Complaint against Defendants Lenovo Group Limited, Lenovo (Shanghai) Electronics Technology Co. Ltd., Lenovo Information Products (Shenzhen) Co. Ltd., LCFC (Hefei) Electronics Technology Co. Ltd. d/b/a LC Future Center and Lenovo Compal Future

Center, Lenovo Centro Tecnológico S. de R.L. de C.V., Lenovo PC HK Ltd., Lenovo (Beijing) Limited, Lenovo (Thailand) Limited, Lenovo Enterprise Technology Company Limited, Lenovo NetApp Technology Limited, Motorola Mobile Communication Technology Ltd., and Motorola (Wuhan) Mobility Technologies Communication Company Limited (collectively, “Lenovo ” or “Defendants”) for patent infringement allege as follows:

THE PARTIES

1. Valtrus is the successor-in-interest to a substantial patent portfolio created by Hewlett Packard Enterprise and its predecessor, subsidiary, and affiliate companies, including Hewlett-Packard Development Company, L.P. (collectively, “HPE”). Valtrus is an Irish entity, duly organized and existing under the laws of Ireland. The address of the registered office of Valtrus is: The Glasshouses GH2, 92 Georges Street Lower, Dun Laoghaire, Dublin A96 VR66, Ireland. HPE’s worldwide corporate headquarters is located in Houston, Texas. One of HPE’s primary U.S. facilities is located in Plano, Texas.

2. KPI is the beneficiary of a trust pursuant to which Valtrus owns, holds, and asserts the Asserted Patents as defined in Paragraph 43. KPI is an Irish entity, duly organized and existing under the laws of Ireland. The address of the registered office of KPI is: The Glasshouses GH2, 92 Georges Street Lower, Dun Laoghaire, Dublin A96 VR66, Ireland.

3. Defendant Lenovo Group Limited is a corporation organized and existing under the laws of China, with its principal place of business at 23rd Floor, Lincoln House, Taikoo Place, 979 King’s Road, Quarry Bay, Hong Kong SAR, China. Lenovo Group Limited is a leading manufacturer and seller of smartphones, computers, servers, and tablets in the world and in the United States. Upon information and belief, Lenovo Group Limited does business in Texas, directly or through intermediaries, and offers its products and/or services, including those accused

herein of infringement, to customers and potential customers located in Texas, including in the Judicial District of the Eastern District of Texas.

4. Lenovo Group Limited, acting in consort with its subsidiaries and affiliates, manufactures and sells computers and handheld devices worldwide, including throughout the United States and within Texas. In 2019, Lenovo was the world's largest supplier of personal computers—accounting for nearly a quarter of global shipments of personal computers. The Lenovo group of companies—collectively known as “The Lenovo Group” or “Lenovo”—have a global reach and have a strong presence in the United States. Lenovo has approximately 77,000, employees worldwide¹, including thousands in the United States,² with many people in the United States involved with management, design, engineering, marketing, and supply chain functions. Lenovo generates more than \$62 billion of goods and services worldwide,³ including over \$19 billion in revenues generated from the Americas⁴, a substantial portion of which is generated in the United States each year. Lenovo maintains one of its dual headquarters in the United States.

5. Officers or executives of Lenovo Group Limited (and/or affiliates acting in concert with Lenovo Group Limited) regularly work from and reside in the United States, including in Texas. These officers or executives include at least Laura Quatela, Paul Rector (Austin, Texas),⁵ Kurt Skaugen, Matthew Zielinski (Austin, Texas),⁶ Doug Fisher, and Arthur Hu. Such officers or

¹<https://www.lenovo.com/us/en/about/?orgRef=https%253A%252F%252Fwww.google.com%252F>

² “Lenovo was named as one of America’s Best Employers for Diversity 2023 by Forbes magazine.” See Lenovo Group Ltd. 2023/2024 Annual Report at 40, available at: <https://investor.lenovo.com/en/publications/reports.php>

³ *Id.*

⁴ See Lenovo Group Ltd. 2023/2024 Annual Report at 187, available at: <https://investor.lenovo.com/en/publications/reports.php>

⁵ See <https://www.linkedin.com/in/paulrector/>

⁶ See <https://www.linkedin.com/in/matthew-zielinski/>

executives are involved in the marketing, distribution, sale, import, and use of the Accused Products (the “Accused Products” include the products and services that infringe one or more claims of one or more patents as set forth in the Counts below) in the United States.

6. Lenovo Group Limited owns or controls, directly or indirectly, each of the Lenovo entities with which it coordinates the infringing sales, offers for sale, import, use, and manufacture of Accused Products in the United States. Lenovo Group Limited operates and manages a global supply chain to develop, manufacture, and deliver accused computer products, server products, and mobile products to the United States, including Texas. At the direction or control of Lenovo Group Limited, Accused Products are packaged, shipped, and sold to customers in the United States. Lenovo Group Limited reports U.S. sales of the Accused Products as its own. Lenovo Group Limited advertises that it manufactures most of its products in its own facilities, rather than through third parties. For example, on its website, Lenovo Group Limited states: “We manufacture the majority of our products in our own facilities—more than most other hardware suppliers. This hybrid model helps us bring new innovations to market efficiently while having greater control over product development and supply chain for advantages in quality, security, and time-to-market. Recently, Gartner ranked us #15 on their list of Top 25 World Class Supply Chains.⁷” On May 28, 2024, Lenovo posted a press release on its website that Gartner ranked it #10 on its list of Supply Chain Top 25 for 2024 report.⁸ In the press release, Lenovo stated: “Today, Lenovo’s global hybrid manufacturing network includes 30+ manufacturing sites spanning 10 markets in the 180 markets Lenovo does business in, including Argentina, Brazil, China, Germany, Hungary,

⁷ See

<https://www.lenovo.com/us/en/about/whoweare/?orgRef=https%253A%252F%252Fwww.google.com%252F>

⁸ See <https://news.lenovo.com/pressroom/press-releases/gartner-supply-chain-top-25-for-2024/>

India, Japan, Mexico, *and the US*. This ensures the supply chain remains resilient and can adapt to any disruption”⁹ (Emphasis added).

7. Defendant Lenovo (Shanghai) Electronics Technology Co., Ltd. (“Lenovo Shanghai”) is a company organized under the laws of the People’s Republic of China. Lenovo Shanghai has an office at No. 68 Building, 199 Fenju Road, Wai Gao Qiao Free Trade Zone, Shanghai, 200131, China, and may be served with process pursuant to the provisions of the Hague Convention. Lenovo Shanghai may also be served with process by serving the Texas Secretary of State, 1019 Brazos Street, Austin, Texas 78701, as its agent for service because it engages in business in Texas but has not designated or maintained a resident agent for service of process in Texas as required by statute. This action arises out of that business.

8. Lenovo Shanghai, an indirect subsidiary of Lenovo Group Limited, has been and is involved in the manufacturing and distribution of accused computer devices, server devices, mobile devices, and other related products. Lenovo Shanghai has been and is involved in the manufacture of Accused Products that are eventually sold to Lenovo PC HK Ltd. or other companies, who then sell the Accused Products to Lenovo (United States) Inc. Lenovo Shanghai knows that the Accused Products it manufactures are intended for the United States market. Lenovo Shanghai has been and is involved in the shipping of Accused Products, such as notebook and desktop computers, to the United States—both directly and through intermediate Lenovo entities. Lenovo Shanghai has been and is involved in shipping at least some of the Accused Products to the United States under the direction or control of Lenovo PC HK Ltd. Lenovo Shanghai’s role within The Lenovo Group of companies includes applying for the necessary

⁹ *Id.*

licenses and authorizations for accused computer products in the United States, including with the FCC, and obtaining the necessary UL certifications.

9. Defendant Lenovo Information Products (Shenzhen) Co. Ltd. (“Lenovo Shenzhen”) is a company organized under the laws of the People’s Republic of China. Lenovo Shenzhen has an office at ISH2 Building, 3 Guangan Road, Futian Free Trade Zone, Shenzhen, 518038, China, and may be served with process pursuant to the provisions of the Hague Convention. Lenovo Shenzhen may also be served with process by serving the Texas Secretary of State, 1019 Brazos Street, Austin, Texas 78701, as its agent for service because it engages in business in Texas but has not designated or maintained a resident agent for service of process in Texas as required by statute. This action arises out of that business.

10. Lenovo Shenzhen, an indirect subsidiary of Lenovo Group Limited, is involved in the manufacturing and distribution of accused computer devices, mobile devices, server devices, and other related products. Specifically, Lenovo has a “manufacturing center” in Shenzhen, China, operated by Lenovo Shenzhen. Lenovo Shenzhen has been and is involved in the manufacture of Accused Products, which then sells to Lenovo PC HK Ltd., who then sells the Accused Products to Lenovo (United States) Inc. Lenovo Shenzhen knows that the Accused Products it manufactures are intended for the United States market. Further, Lenovo Shenzhen has been and is involved in the shipping of Accused Products, such as notebook and desktop computers, to the United States—both directly and through intermediate Lenovo entities. Lenovo Shenzhen has been and is involved in shipping at least some of the Accused Products to the United States under the direction or control of Lenovo Group Limited, and/or Lenovo PC HK Ltd.

11. LCFC (Hefei) Electronics Technology Co., Ltd. d/b/a LC Future Center and Lenovo Compal Future Center (“LCFC Hefei”) is a company organized under the laws of a foreign

jurisdiction. LCFC Hefei has an office at 7th Floor, No. 780, Bei'an Road, Zhongshan District, Taipei City 10491, Taiwan, and may be served with process pursuant to the provisions of the Hague Convention. LCFC Hefei may also be served with process by serving the Texas Secretary of State, 1019 Brazos Street, Austin, Texas 78701, as its agent for service because it engages in business in Texas but has not designated or maintained a resident agent for service of process in Texas as required by statute. This action arises out of that business.

12. LCFC Hefei, an indirect subsidiary of Lenovo Group Limited, is involved in the manufacturing and distribution of accused computer devices, mobile devices, server devices, and other related products. Over 60% of Lenovo laptops are manufactured by LCFC Hefei. Specifically, Lenovo has a “manufacturing center” in Hefei, China, operated by LCFC Hefei. LCFC Hefei has been and is involved in the manufacture of Accused Products that are eventually sold to Lenovo PC HK Ltd., which then sells the Accused Products to Lenovo (United States) Inc. LCFC Hefei knows that the Accused Products it manufactures are intended for the United States market. Further, LCFC Hefei has been and is involved in the shipping of Accused Products, such as notebook computers, to the United States—both directly and through intermediate Lenovo entities. LCFC Hefei has been and is involved in shipping at least some of the Accused Products to the United States under the direction or control of Lenovo Group Limited and/or Lenovo PC HK Ltd.

13. Lenovo Centro Tecnológico S. de R.L. de C.V. (“Lenovo Centro”) is a company organized under the laws of Mexico. Lenovo Centro has an office at No. 316, Boulevard Escobedo, Apodaca Technology Park, Apodaca, Nuevo León, P.O. 66600, México, and may be served pursuant to the provisions of the Hague Convention. Lenovo Centro may also be served with process by serving the Texas Secretary of State, 1019 Brazos Street, Austin, Texas 78701, as

its agent for service because it engages in business in Texas but has not designated or maintained a resident agent for service of process in Texas as required by statute. This action arises out of that business.

14. Lenovo Centro, an indirect subsidiary of Lenovo Group Limited, is involved in the manufacturing, assembly, and testing of accused personal computers and other related products. Specifically, Lenovo has a “manufacturing center” in Monterrey, Mexico, operated by Lenovo Centro. Lenovo Centro has been and is involved in the manufacture of Accused Products that it sells to Lenovo PC HK Ltd., which then sells the Accused Products to Lenovo (United States) Inc. Lenovo Centro knows that the Accused Products it manufactures are intended for the United States market. Further, Lenovo Centro has been and is involved in the shipping of Accused Products, such as desktop computers, to the United States—both directly and through intermediate Lenovo entities. Lenovo Centro has been and is involved in shipping at least some of the Accused Products to the United States under the direction or control of Lenovo Group Limited and/or Lenovo PC HK Ltd.

15. Lenovo PC HK Ltd. (“Lenovo PC HK”) is a company organized under the laws of Hong Kong SAR. Lenovo PC HK has an office at 23rd Floor, Lincoln House, Taikoo Place, 979 King’s Road, Quarry Bay, Hong Kong SAR, China, and may be served pursuant to the provisions of the Hague Convention. Lenovo PC HK may also be served with process by serving the Texas Secretary of State, 1019 Brazos Street, Austin, Texas 78701, as its agent for service because it engages in business in Texas but has not designated or maintained a resident agent for service of process in Texas as required by statute. This action arises out of that business.

16. Lenovo PC HK, an indirect subsidiary of Lenovo Group Limited, is involved in the procurement, marketing, distribution, and sale of accused computer devices, mobile devices,

server devices, and other related products. In particular, at the direction and control of Lenovo Group Limited, Lenovo PC HK procures, makes, sells, offers for sale, imports, and uses Accused Products manufactured for the United States by Lenovo entities, such as LCFC Hefei and Lenovo Centro. That role includes providing manufacturers with designs and specifications of accused computer products destined for the United States.

17. Further, Lenovo PC HK has been and is involved in the shipping of Accused Products, such as notebook and desktop computers, to the United States—both directly and through intermediate Lenovo entities. Lenovo PC HK also has sold and sells accused Lenovo products, such as notebook and desktop computers, to at least one U.S. distributor—Lenovo (United States) Inc. — which then re-sells them to Lenovo customers residing in the United States. Lenovo PC HK does the above activities at the instruction and under the supervision of Lenovo Group Limited. Defendants are aware that the Accused Products sold in the United States, such as to Lenovo (United States) Inc., are distributed throughout the United States, including in Texas.

18. Lenovo (Beijing) Limited (“Lenovo Beijing”) is a company organized under the laws of China. Lenovo Beijing has an office at No. 6, Chuangye Road, Shangdi, Haidan District, Beijing 100085, China, and may be served with process pursuant to the provisions of the Hague Convention. Lenovo Beijing may also be served with process by serving the Texas Secretary of State, 1019 Brazos Street, Austin, Texas 78701, as its agent for service because it engages in business in Texas but has not designated or maintained a resident agent for service of process in Texas as required by statute. This action arises out of that business.

19. Lenovo Beijing is a subsidiary of Lenovo Group Limited. Upon information and belief, Lenovo Beijing acts in concert with, and under the direction and control of, Lenovo Group Limited and/or Lenovo PC HK to sell, offer for sale, import, use, and make the Accused Products

in the United States. For example, Lenovo Beijing is the registrant of www.lenovo.com, through which the Defendants sell the Accused Products directly to customers in the United States. Lenovo Beijing further operates the “Beijing Data Center to support Lenovo global core business.” Upon information and belief, Lenovo Beijing, operating in concert with Defendants, host www.lenovo.com from the “Beijing Data Center.”

20. Defendant Lenovo Enterprise Technology Company Limited (“Lenovo Enterprise”) is a corporation organized and existing under the laws of China, with its principal place of business at 23rd Floor, Lincoln House, Taikoo Place, 979 King’s Road, Quarry Bay, Hong Kong SAR, China,¹⁰ and may be served with process pursuant to the provisions of the Hague Convention. Upon information and belief, Lenovo Enterprise acts in consort with, and under the direction and control of, Lenovo Group Limited and/or other Lenovo Group Limited subsidiaries/affiliates to sell, offer for sale, import, use, and make the Accused Products in the United States, including computer and server products.¹¹

21. Defendant Lenovo NetApp Technology Limited (“Lenovo NetApp”) is a corporation organized and existing under the laws of China, with its principal place of business at Lenovo Innovation Park (Tianjin), Jingsan Road, Airport Economic Development Zone, Dongli District, Tianjin City, China,¹² and may be served with process pursuant to the provisions of the Hague Convention. Upon information and belief, Lenovo NetApp acts in consort with, and under

¹⁰ See

<https://eulerpool.com/en/company/HK/Lenovo%20Enterprise%20Technology%20Company%20Limited,Hong%20Kong,2811034>

¹¹ See LGL 2023/2024 Annual Report at 260, available at: <https://investor.lenovo.com/en/publications/reports.php>

¹² See

<https://eulerpool.com/en/company/HK/Lenovo%20Enterprise%20Technology%20Company%20Limited,Hong%20Kong,2811034>

the direction and control of, Lenovo Group Limited and/or other Lenovo Group Limited subsidiaries/affiliates, such as Lenovo Beijing, to sell, offer for sale, import, use, and make the Accused Products in the United States, including computer and server products.¹³

22. Motorola Mobile Communication Technology Ltd. (“Motorola Comm.”) is a wholly-owned subsidiary of Lenovo Group Limited organized under the laws of China. Upon information and belief, Motorola Comm. acts in concert with, and under the direction and control of, Lenovo Group Limited and/or other Lenovo Group Limited subsidiaries/affiliates to sell, offer for sale, import, use, and make the Accused Products in the United States. Motorola Comm. may be served with process by serving the Texas Secretary of State, 1019 Brazos Street, Austin, Texas 78701, as its agent for service because it engages in business in Texas but has not designated or maintained a resident agent for service of process in Texas as required by statute. This action arises out of that business.

23. Motorola (Wuhan) Mobility Technologies Communication Company Limited (“Motorola Wuhan”) is a wholly-owned subsidiary of Lenovo Group Limited, organized under the laws of China, with a principal place of business located at No. 19, Gaoxin 4th Road, Donghu New Technology Development Zone Wuhan, Hubei, 430205 China and may be served with process pursuant to the provisions of the Hague Convention. Motorola Wuhan may also be served with process by serving the Texas Secretary of State, 1019 Brazos Street, Austin, Texas 78701, as its agent for service because it engages in business in Texas but has not designated or maintained a resident agent for service of process in Texas as required by statute. This action arises out of that business. Upon information and belief, Motorola Wuhan acts in concert with, and under the

¹³ See LGL 2023/2024 Annual Report at 262, available at: <https://investor.lenovo.com/en/publications/reports.php>

direction and control of, Lenovo Group Limited and/or other Lenovo Group Limited subsidiaries/affiliates to sell, offer for sale, import, use, and make the Accused Products in the United States, including mobile products.¹⁴

24. The Defendants identified in paragraphs 3 through 23 above are companies that together — with their affiliates — comprise one of the world’s leading manufacturers of computers and computer-related products. The Defendants design, manufacture, use, import into the United States, sell, and/or offer for sale in the United States computer devices, mobile devices, server devices, and other related products. Lenovo’s devices are marketed, offered for sale, and/or sold throughout the United States, including within this District.

25. The Defendants named above and their affiliates are part of the same corporate structure and distribution chain for the making, importing, offering to sell, selling, and using of the Accused Products in the United States, including in the State of Texas generally and this Judicial District in particular. Defendants engage in coordinated and concerted action to direct the Accused Products throughout the United States, including Texas.

26. The Defendants named above and their affiliates share the same management, common ownership, advertising platforms, facilities, distribution chains and platforms, and accused product lines and products involving related technologies.

27. In the promotional materials, manuals, guides, terms of use, sales agreements, warranties, or similar documentation related to the Accused Products, the Defendants regularly omit which specific Lenovo company or entity is responsible for the documents or associated products, or instead identify “Lenovo” or the “Lenovo Group ”. As a result, customers of the

¹⁴ See Lenovo Group Limited 2023/2024 Annual Report at 262, available at: <https://investor.lenovo.com/en/publications/reports.php>.

Accused Products understand that Lenovo Group Limited or the Lenovo Group as a whole, makes and sells the Accused Products.

28. Lenovo Group Limited and Lenovo as a whole, holds itself out as the entity that manufactures, sells, offers to sell, imports, and uses the Accused Products in the United States, including Texas. For example, the privacy statement of www.lenovo.com states that “[t]his privacy statement applies to data collected through websites owned and operated by Lenovo Group Ltd. and its affiliated group companies (“Lenovo”).”¹⁵ Lenovo sells and offers for sale products through its U.S. website, and “Lenovo” (*i.e.* “Lenovo Group Ltd. and its affiliated group companies”) imposes the terms of a “Lenovo Sales Agreement” on all customers, including customers within the United States and Texas.¹⁶ Lenovo Group Limited and Lenovo as a whole, sell the Accused Products directly to customers in the United States, including within Texas, through www.lenovo.com/us/en.

29. The Defendants named above and their affiliates operate as a unitary business venture and are both jointly and severally liable for the acts of patent infringement alleged herein.

¹⁵ <https://www.lenovo.com/us/en/privacy/>. *See also* <https://www.lenovo.com/us/en/legal/> (“The following are terms of a legal agreement between you and Lenovo (“we,” “us,” or “our”)”

¹⁶ <https://www.lenovo.com/medias/Sales-Terms-and-Conditions-US.html?context=bWFzdGVyfHJvb3R8MTMzNzV8dGV4dC9odG1sfGg3MC9oYWMvOTQ0MTA1NTgwMTM3NC5odG1sfDgwM2RjYzIxMzNhYWYzOTJiNGEwZjU1ZjFhMWZkOGM5M2JhYzVmYTkwNzQ2OTk0ZWE5NjVhOWZiMWYwNzdhZmE>

JURISDICTION AND VENUE

30. This is an action for patent infringement arising under the patent laws of the United States, 35 U.S.C. §§ 1, *et seq.* This Court has jurisdiction over this action pursuant to 28 U.S.C. §§ 1331 and 1338(a).

31. This Court has personal jurisdiction over Defendants. Defendants regularly conduct business and have committed acts of patent infringement and/or have induced acts of patent infringement by others in this Judicial District and/or have contributed to patent infringement by others in this Judicial District, the State of Texas, and elsewhere in the United States. Defendants, directly and/or through subsidiaries or intermediaries, have committed and continue to commit acts of infringement in this District by, among other things, making, using, importing, offering to sell, and/or selling products that infringe the Patents-in-Suit as defined in paragraph 43. Courts in Texas, including within this Judicial District, have concluded that at least Lenovo Group Limited is subject to personal jurisdiction in the State of Texas. *See ACQIS LLC v. Lenovo Grp. Ltd.*, 572 F. Supp. 3d 291, 307 (W.D. Tex. 2021) (“this Court finds that the exercise of personal jurisdiction over [Lenovo Group Limited] is both reasonable and fair.”); *see also AX Wireless LLC v. Lenovo Grp. Ltd.*, No. 2:22-cv-00280-RWS-RSP, Dkt. No. 110 (report and recommendation) (E.D. Tex. Sept. 6, 2023) (“exercising personal jurisdiction [over Lenovo Grp. Ltd.] would not offend traditional notions of fair place and substantial justice.”); *Eireog Innovations Ltd. v. Lenovo Grp. Ltd.*, No. 2:24-CV-00239-JRG, 2024 WL 4519763 (E.D. Tex. Oct. 17, 2024); *Universal Connectivity Techs. Inc. v. Lenovo Grp. Ltd.*, No. 2:23-CV-00449-JRG, 2024 WL 4519760 (E.D. Tex. Oct. 17, 2024); *Truesight Commc’ns LLC v. Lenovo Grp. Ltd.*, No. 2:24-CV-00031-JRG, 2024 WL 5110057 (E.D. Tex. Dec. 13, 2024).

32. Lenovo induces its subsidiaries, affiliates, retail partners, and customers to make, use, sell, offer for sale, and/or import throughout the United States, including within this Judicial District, infringing products and places such products into the stream of commerce via established distribution channels knowing or understanding that such products would be sold and used in the United States, including in the Eastern District of Texas. Paragraphs 1 through 31 are incorporated by reference as if fully set forth herein. Lenovo purposefully directs the infringing products identified herein into established distribution channels within this District and the U.S. nationally. For example, Lenovo sells and offers to sell the infringing products through its websites, [Lenovo.com](https://www.lenovo.com)¹⁷ and [Motorola.com](https://www.motorola.com), which may be accessed throughout the United States, the State of Texas, and this Judicial District. Additionally, Lenovo has authorized sellers and sales representatives that offer for sale and sell the infringing products throughout the State of Texas and to consumers throughout this District, including at the following locations in this District: Best Buy, 422 West Loop 281, Suite 100, Longview, Texas 75605; Costco Wholesale, 3650 West University Drive, McKinney, Texas 75071; Office Depot, 422 West Loop 281, Suite 300, Longview, Texas 75605; Target, 3092 North Eastman Road, Suite 100, Longview, Texas 75605; Wal-Mart, 1701 East End Boulevard North, Marshall, Texas 75670.

33. Lenovo maintains regular and established places of business in this Judicial District, the State of Texas, and elsewhere in the United States, including a Sales Office in Fort

¹⁷ The Lenovo website is maintained by Lenovo Beijing Limited. *See* <https://www.whois.com/whois/lenovo.com>

Worth, Texas.¹⁸ Lenovo also advertises for jobs in the State of Texas.¹⁹ Lenovo also maintains Lenovo Authorized Service Providers (ASPs) in this Judicial District, the State of Texas, and elsewhere in the United States.

Location Results

We found 14 locations

- Lenovo Authorized Service Providers (ASPs) offer limited warranty service support.
- Prior to visiting an ASP location, you must call first to ensure that the ASP can provide the assistance you require.
- If a repair case was initiated with Lenovo where part(s) were shipped to your house/office, you must wait for Lenovo to contact you and resolve. You can also contact Lenovo at 1-855-253-6686 option #1. ASPs are not authorized to replace parts or repair machines for customers who have initiated service through Lenovo.
- Authorized Service Providers may charge for out of scope services. Lenovo does not regulate and is not responsible for the prices charged.

COMPUTERLAND NET TECH E TEX

23.30 Miles

75604,1614 EAST FAIRMONT, LONGVIEW, Texas, United States

Website: <http://www.cynergytech.com>

Premises Type: Walk in; Retail; Corporate

QUALITY HIGH-TECH LLC

139.23 Miles

75081-6743, 1910 Firman Dr. Suite 120, Richardson, Texas, United States

Website: <http://www.ghl.com/>

Premises Type: Walk in; Retail; Corporate

Micro Center DLSE Store 134

141.05 Miles

75243-1007, 13929 North Central

¹⁸https://www.lenovo.com/us/en/about/locations/?orgRef=https%253A%252F%252Fwww.google.com%252F&cid=us:sem|se|google|pmax_smb_pcs|||18337003604||shopping|mix|commercialconsumer&gad_source=1&gclid=CjwKCAjwvvmzBhA2EiwAtHVrb-yDPR8mEb0OOJoQq7k6lSpSZJTL4W1Ucy0kqq6ZBDwyz4H5zd7rSBoc6NQQAvD_BwE

¹⁹ See, e.g., https://jobs.lenovo.com/en_US/careers/JobDetail/NA-Workstation-Channel-Account-Manager/58039. (“This is a remote role based in Austin, TX, candidate must reside in territory.”).

Lenovo Service Center FAQs

1. What is Lenovo Service?

Lenovo Service refers to the range of support options and services provided by Lenovo to assist customers with their Lenovo devices. It includes technical support, repairs, warranty coverage, and other assistance aimed at ensuring the optimal performance of Lenovo products.

2. What is a Lenovo Service Center?

A Lenovo Service Center is an authorized facility where customers can seek professional assistance for their Lenovo devices. These centers employ trained technicians who specialize in Lenovo products and provide services such as diagnostics, repairs, hardware upgrades, and software installations.

3. What is the difference between a Lenovo Service Center and a Lenovo Authorized Service Center?

A Lenovo Service Center and a Lenovo Authorized Service Center serve the same purpose of providing support for Lenovo devices. However, an Authorized Service Center has received certification from Lenovo to meet specific standards and requirements set by the company. This certification ensures that the service center is qualified to deliver reliable and high-quality service.

4. How can I find a Lenovo Service Center near me?

To find a Lenovo Service Center near you, you can visit the [Lenovo support website](#) and use the service center locator tool. Enter your location or zip code, and the tool will provide a list of authorized service centers in your area.

5. What services are provided by Lenovo Service Centers?

Lenovo Service Centers offer a wide range of services for Lenovo devices, including diagnostics, repairs, hardware upgrades, software installations, and warranty support. They can assist with both in-warranty and out-of-warranty issues, ensuring that customers receive the necessary support for their devices.

Lenovo provides a range of support services and authorized service centers to ensure that customers receive the necessary assistance for their Lenovo devices. Whether it's finding a service center, accessing general support, or utilizing self-service diagnostics, Lenovo is dedicated to providing reliable and comprehensive service options for its customers.

20

Defendants also maintain authorized resellers of their products in this Judicial District, in the State of Texas, and throughout the United States.

²⁰ See, e.g., <https://support.lenovo.com/us/en/lenovo-service-provider> Displaying result for “Warranty and Support Provider” for “Home” within 500 miles of 200 W Houston Street, Marshall, TX 75670, USA. (“A Lenovo Service Center is an authorized facility where customers can seek professional assistance for their Lenovo devices. These centers employ trained technicians who specialize in Lenovo products and provide services such as diagnostics, repairs, hardware upgrades, and software installations.”).

Purpose

* Need to find a reseller or a service provider

☒ Reseller ¹

☐ Warranty and Support Provider ¹

* Need assistance with products for home or for business

☒ Business

Location

☐ Use my current location

Address*

200 W Houston St, Marshall, TX 75670, USA

* Measured in

☒ Miles

☐ Kilometers

Distance*

500

SEARCH

Location Results

We found 23 locations

- Lenovo Authorized Service Providers (ASPs) offer limited warranty service support.
- Prior to visiting an ASP location, you must call first to ensure that the ASP can provide the assistance you require.
- If a repair case was initiated with Lenovo where part(s) were shipped to your house/office, you must wait for Lenovo to contact you and resolve. You can also contact Lenovo at 1-855-253-6686 option #1. ASPs are not authorized to replace parts or repair machines for customers who have initiated service through Lenovo.
- Authorized Service Providers may charge for out of scope services. Lenovo does not regulate and is not responsible for the prices charged.

Medwed Enterprises, Inc.

Lenovo 360 Intelligent Devices Partner 2023 Gold

Lenovo 360 Infrastructure Solutions Partner 2023 Gold

139.85 Miles

1401 N. Central Expressway
#320, Richardson, Texas, United States

Phone Number: (972) 644-1992

Reliable It LLC

Lenovo 360 Intelligent Devices Partner 2023 Gold

Lenovo 360 Infrastructure Solutions Partner 2023 Gold

141.14 Miles

1201 3rd St Suite 100, Alexandria, Louisiana, United States

Phone Number: (949) 419-8000

21

At these various locations, Lenovo maintains authorized sellers, sales representatives, and customer service agents that offer, sell, and service Defendants' products in this Judicial District, the State of Texas, and elsewhere in the United States including, but not limited to, Computerland Net Tech E Tex at 1614 East Fairmont Longview, Texas 75604.

34. Furthermore, Lenovo Group Limited provides "Hardware Maintenance Manuals" for the Accused Products in the United States which bear a "Lenovo" copyright without specifying any particular Lenovo entity, and which are delivered pursuant to a General Services

²¹ *Id.* Search results for "Reseller" for "Business" within 500 miles of 200 West Houston Street, Marshall, TX 75670, USA.

Administration “GSA” contract. These manuals specify that they are “printed in China.” These English language manuals demonstrate Lenovo’s knowledge and intent that the Accused Products are to be sold throughout the United States, including Texas.

35. Venue is proper in this Judicial District pursuant to 28 U.S.C. § 1391 because, among other things, Defendants are not residents in the United States, and thus may be sued in any judicial district pursuant to 28 U.S.C. § 1391(c)(3).

36. Defendants are subject to this Court’s jurisdiction pursuant to due process and/or the Texas Long Arm Statute due at least to their substantial business in this State and Judicial District, including (a) at least part of their past infringing activities, (b) regularly doing or soliciting business in Texas, and/or (c) engaging in persistent conduct and/or deriving substantial revenue from goods and services provided to customers in Texas. In addition, or in the alternative, this Court has personal jurisdiction over at least Lenovo Group Limited pursuant to Fed. R. Civ. P. 4(k)(2) because: (1) it has substantial contacts with the United States and committed and/or induced acts of patent infringement in the United States; and (2) it is not subject to jurisdiction in any state’s courts of general jurisdiction.

PATENTS-IN-SUIT

37. On November 17, 2015, the United States Patent and Trademark Office duly and legally issued U.S. Patent No. 9,191,989 (the “’989 Patent”) entitled “Apparatus and Associated Method, for Controlling Connectivity of a Computer Device with a Computer Network”. A true and correct copy of the ’989 Patent is available at: <https://patentimages.storage.googleapis.com/3d/8c/ef/31d174aed4de3d/US9191989.pdf>.

38. On December 11, 2012, the United States Patent and Trademark Office duly and legally issued U.S. Patent No. 8,332,930 (the “’930 Patent”) entitled “Secure Use of User Secrets

on a Computing Platform”. A true and correct copy of the ’930 Patent is available at: <https://patentimages.storage.googleapis.com/36/f1/b6/68f62bbbe02194/US8332930.pdf>.

39. On April 19, 2011, the United States Patent and Trademark Office duly and legally issued U.S. Patent No. 7,930,539 (the “’539 Patent”) entitled “Computer System Resource Access Control”. A true and correct copy of the ’539 Patent is available at: <https://patentimages.storage.googleapis.com/4f/8d/8e/e944beee8c2f28/US7930539.pdf>.

40. On December 29, 2009, the United States Patent and Trademark Office duly and legally issued U.S. Patent No. 7,640,332 (the “’332 Patent”) entitled “System and Method for Hot Deployment/Redeployment in Grid Computing Environment”. A true and correct copy of the ’332 Patent is available at: <https://patentimages.storage.googleapis.com/ff/8a/a0/8a59d0b8c2477c/US7640332.pdf>.

41. On June 6, 2006, the United States Patent and Trademark Office duly and legally issued U.S. Patent No. 7,057,509 (the “’509 Patent”) entitled “Monitoring an Object with Identification Data and Tracking Data”. A true and correct copy of the ’509 Patent is available at: <https://patentimages.storage.googleapis.com/a6/a7/69/01682103e93dd7/US7057509.pdf>.

42. On October 10, 2006, the United States Patent and Trademark Office duly and legally issued U.S. Patent No. 7,120,832 (the “’832 Patent”) entitled “Storage Device Performance Monitor”. A true and correct copy of the ’832 Patent is available at: <https://patentimages.storage.googleapis.com/07/09/dd/ac3e2a777310b0/US7120832.pdf>.

43. Plaintiffs are the sole and exclusive owners of all legal right, title, and interest in the ’989 Patent, the ’930 Patent, the ’539 Patent, the ’332 Patent, the ’509 Patent, and the ’832 Patent (collectively, the “Patents-in-Suit” or “Asserted Patents”), and hold the exclusive right to take all actions necessary to enforce their rights to the Patents-in-Suit, including the filing of this

patent infringement lawsuit. Plaintiffs also have the right to recover all damages for past, present, and future infringement of the Patents-in-Suit and to seek injunctive relief as appropriate under the law.

FACTUAL ALLEGATIONS

44. The '989 Patent generally relates to technology for controlling the communication connectivity of a computer device between several networks. The technology described in the '989 Patent was developed by Hans F. Valine at Hewlett-Packard Development Company, L.P. For example, the technology is implemented in infringing Lenovo smartphones, including Motorola branded smartphones, with the ability to use Airplane Mode.

45. The '930 Patent generally relates to technology including a computing platform for secure use of a trusted user input, the computing platform comprising a user input device, a first isolated operating environment for execution of applications requiring use of trusted user input, a second isolated operating environment adapted for secure processing of information relating to a user, and a third isolated operating environment adapted for secure communication with the user input device. The technology described in the '930 Patent was developed by Will Burton and Boris Balacheff at Hewlett-Packard Development Company, L.P. For example, the technology is implemented in Lenovo devices, such as Motorola branded smartphones, having a secure authentication and/or biometric sensor functionality for accessing certain features, programs, and/or software, such as third-party applications.

46. The '539 Patent generally relates to technology implemented in a computer system having a plurality of resources, where the computer system receives a request from a software program to access one of the resources and determines if the resource is a protected resource. If the resource is a protected resource, the computer system, if operating in a protected mode or non-

protected mode, determines whether access to the resource should be granted. The technology described in the '539 Patent was developed by Donald C. Soltis, Jr., Rohit Bhatia, and Eric R. DeLano at Hewlett-Packard Development Company, L.P. For example, the technology is implemented in Lenovo devices equipped with AMD multicore processors, AMD EPYC processors, AMD Ryzen processors, and AMD Athlon processors, such as Lenovo servers, desktops, laptops, and workstations.

47. The '332 Patent generally relates to technology for hot deployment and/or redeployment in a grid computing environment, where the grid computing environment adds a new version of an application from a repository server, determining which grid nodes are running the application affected by the new version of the software, notifying the client application manager associated with one or more of the affected grid nodes, and hot deploying/redeploying the new version of the application to the affected grid nodes. The technology described in the '332 Patent was developed by Sanjay Dahiya at Hewlett-Packard Development Company, L.P. For example, the technology is used by Lenovo and/or is implemented in systems using Kubernetes for rolling updates, such as offered by Lenovo's Infrastructure Solutions Group including, but not limited to, ThinkSystem and ThinkAgile.

48. The '509 Patent generally relates to technology for a monitoring apparatus and method comprising a first interface for receiving identification data from an identification system, a second interface for receiving tracking data from a tracking system, a merging unit for merging said data to form monitoring data, and an analyzer unit for processing the monitoring data. The technology described in the '509 Patent was developed by Giovanni Gualdi, Cyril Brignone, and Salil Pradhan at Hewlett-Packard Development Company, L.P. For example, the technology is used by Lenovo and/or is implemented in Lenovo products that feature location sharing, such as

Motorola branded phones through the use of Google Maps.

49. The '832 Patent generally relates to technology for monitoring performance of a storage device by intercepting communications between a computer system and said storage device, analyzing the intercepted communications, and reallocating at least some of said data on said storage device to enhance the performance of the storage device. The technology described in the '832 Patent was developed by Kevin Collins and Michael P. Fleischmann at Hewlett-Packard Development Company, L.P. For example, the technology is used by Lenovo and/or is implemented in Lenovo servers, storage systems, data centers, e-commerce, other systems that use Apache Kafka Streams using RocksDB Universal Compaction and also systems using Apache Cassandra.

50. Lenovo has infringed the '832 Patent and the '509 Patent by making, using, selling, offering to sell, and/or importing, and by actively inducing others to make, use, sell, offer to sell, and/or import products including, but not limited to, servers, storage devices, smartphones, and other infringing technology.

51. Lenovo has infringed and is continuing to infringe the '989 Patent, the '930 Patent, the '539 Patent, and the '332 Patent, by making, using, selling, offering to sell, and/or importing and by actively inducing others to make, use, sell, offer to sell, and/or import, products including, but not limited to, smartphones, computers, servers, and other infringing technology.

COUNT I
(Infringement of the '989 Patent)

52. Paragraphs 1 through 51 are incorporated by reference as if fully set forth herein.

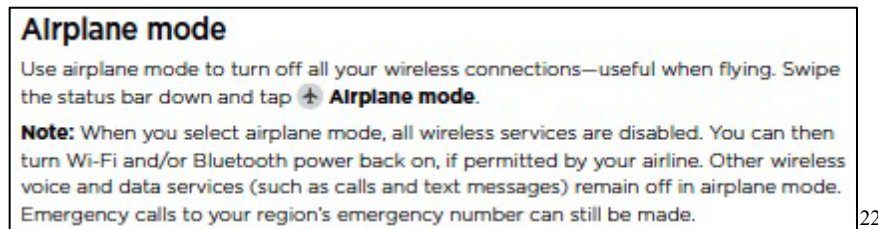
53. Plaintiffs have not licensed or otherwise authorized Defendants to make, use, offer for sale, sell, or import any products that embody the inventions of the '989 Patent.

54. Defendants have and continue to directly infringe the '989 Patent, either literally or

under the doctrine of equivalents, without authority and in violation of 35 U.S.C. § 271, by making, using, offering to sell, selling, and/or importing into the United States products that satisfy each and every limitation of one or more claims of the '989 Patent. Such infringing products include Lenovo smartphones, such as the Motorola One 5G Ace, with the ability to use Airplane Mode.

55. For example, Defendants have and continue to directly infringe at least claim 12 of the '989 Patent by making, using, offering to sell, selling, and/or importing into the United States products that include Lenovo smartphones including, but not limited to, the Motorola One 5G Ace, among other Lenovo products. On information and belief, Defendants' direct infringement includes, but is no limited to, testing in the United States.

56. The Motorola One 5G Ace performs a method for providing selectable communication connectivity of a computer device with a first network and with a second network. For example, the Motorola One 5G Ace can be placed in selectable communication connectivity with Wi-Fi, and cellular networks.

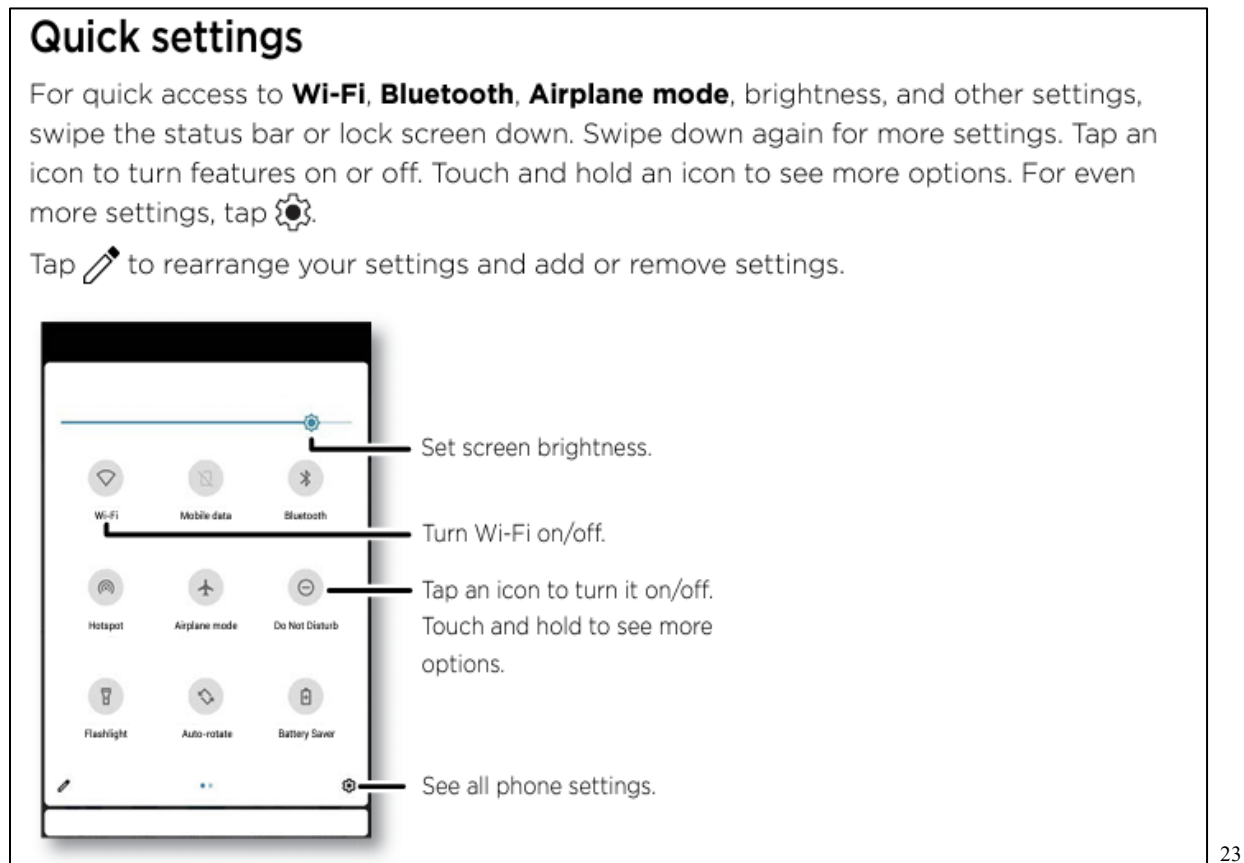


22

57. The Motorola One 5G Ace performs the step of detecting a user selection of which, if any, of the first network and the second network to form a selected network with which to be

²² Motorola One 5G Ace User Guide. available at: https://en-us.support.motorola.com/ci/fattach/get/74876372/1625760959/redirect/1/session/L2F2LzEvdGltZS8xNzM0NTc5NzY1L2dlbi8xNzM0NTc5NzY1L3NpZC9mVXVTdIFKJTdFMFg0cXU5cFZSdjREYkZ4ajVuc1pkQVFUeVBkeVhLVE1nUEJWVEw0MTdPdIZvNDNRyYk1Xd09sMG9NOWlOOHhMbIN0N0dlbEp0M3dfbmlZdXA0TWtRaUtRNDRwNnB1ZVZsTkMybGlKNEhUUENFNXXNrdyUyMSUyMQ==/filename/motorola+one+5G+ace+%26+motorola+one+5G+UW+ace.NA+Retail.UG.en_US.SSC8D10322-B.pdf

placed in communication connectivity. For example, the Motorola One 5G Ace provides a user interface for a user to select Airplane Mode, which prevents users from using cellular data, while permitting a user to select Wi-Fi.



23

The Motorola One 5G Ace performs the step of enabling the communication connectivity with the selected network while preventing communication connectivity with a non-selected network while the communication connectivity with the selected network is enabled. For example, while

²³ Motorola One 5G Ace User Guide, available at: https://en-us.support.motorola.com/ci/fattach/get/74876372/1625760959/redirect/1/session/L2F2LzEvdGltZS8xNzM0NTc5NzY1L2dlbi8xNzM0NTc5NzY1L3NpZC9mVXVTdlFKJTdFMFg0cXU5cFZSdjREYkZ4ajVuc1pkQVFUeVBkeVhLVE1nUEJWVEw0MTdPdZvNDNrYk1Xd09sMG9NOWlOOHhMbIN0N0dlbEp0M3dfbm1ZdXA0TWtRaUtRNDRwNnB1ZVZsTkMybGlKNEhUUE NFNXNrdyUyMSUyMQ==/filename/motorola+one+5G+ace+%26+motorola+one+5G+UW+ac e.NA+Retail.UG.en_US.SSC8D10322-B.pdf

in Airplane Mode, the Motorola One 5G Ace cannot connect to cellular data, but it can connect to Wi-Fi.

58. Lenovo directly infringes the '989 Patent through promotional demonstrations, testing, or repairs, and/or instructional guidance of the Motorola Razr, performing at least the steps of claim 12.

59. Defendants have and continue to indirectly infringe one or more claims of the '989 Patent by knowingly and intentionally inducing others, including Lenovo customers and end-users, to directly infringe, either literally or under the doctrine of equivalents, by making, using, offering to sell, selling, and/or importing into the United States products that include the infringing technology.

60. Defendants, with knowledge²⁴ that these products, or the use thereof, infringe the '989 Patent at least as of July 27, 2021,²⁵ knowingly and intentionally induced, and continues to knowingly and intentionally induce, direct infringement of the '989 Patent by providing these products to end-users for use in an infringing manner. Additionally, on information and belief, Defendants have adopted a policy of not reviewing the patents of others, including specifically those related to Defendants' specific industry, thereby remaining willfully blind to the '989 Patent

²⁴ Lenovo cited directly to the '989 Patent against its own U.S. Patent Application No. 17/707,229, which was published on October 5, 2023, with Publication No. US 2023/0319151 A1, and issued as U.S. Patent No. 11,785,101.

²⁵ Plaintiffs sent Defendant Lenovo Group Limited correspondence on July 27, 2021, notifying them of its infringement of the '989 Patent, among other patents. Plaintiffs sent follow-up correspondence to Defendant Lenovo Group Limited regarding the original letter and additional patents on September 2, 2021. Thereafter, on September 30, 2021, Plaintiffs received an email from licensing@lenovo.com, acknowledging receipt of the September 2, 2021 letter and indicating that Defendant Lenovo Group Limited was willing to discuss the contents thereof. Through at least November 2024, Plaintiffs and Defendant Lenovo Group Limited negotiated a license to various patents, including the Asserted Patents. No license was ever entered. Therefore, at least Defendant Lenovo Group Limited has been on notice of its infringement of the '989 Patent since no later than July 27, 2021.

at least as early as the issuance of the '989 Patent.

61. Defendants have induced infringement by others, including end-users, with the intent to cause infringing acts by others or, in the alternative, with the belief that there was a high probability that others, including end-users, infringe the '989 Patent, but while remaining willfully blind to the infringement. Defendants have and continue to induce infringement by their customers and end-users by supplying them with instructions on how to operate the infringing technology in an infringing manner, while also making publicly available information on the infringing technology via Defendants' website, product literature and packaging, and other publications.²⁶

62. Plaintiffs have suffered damages as a result of Defendants' direct and indirect infringement of the '989 Patent in an amount to be proven at trial.

63. Plaintiffs have suffered, and will continue to suffer, irreparable harm as a result of Defendants' infringement of the '989 Patent, for which there is no adequate remedy at law, unless Defendants' infringement is enjoined by this Court.

64. Defendants have committed and continue to commit acts of infringement that Defendants actually knew or should have known constituted an unjustifiably high risk of infringement of at least one valid and enforceable claim of the '989 Patent. Defendants' direct and indirect infringement of the '989 Patent has been and continues to be willful, intentional, deliberate, and/or in conscious disregard of rights under the patent. Plaintiffs are entitled to an award of treble damages, reasonable attorney fees, and costs in bringing this action.

COUNT II
(Infringement of the '930 Patent)

²⁶ See, e.g., Motorola One 5G Ace User Guide (HTML), available at: https://en-us.support.motorola.com/app/answers/detail/a_id/156861/~motorola-one-5g-ace---user-guide-%28html%29

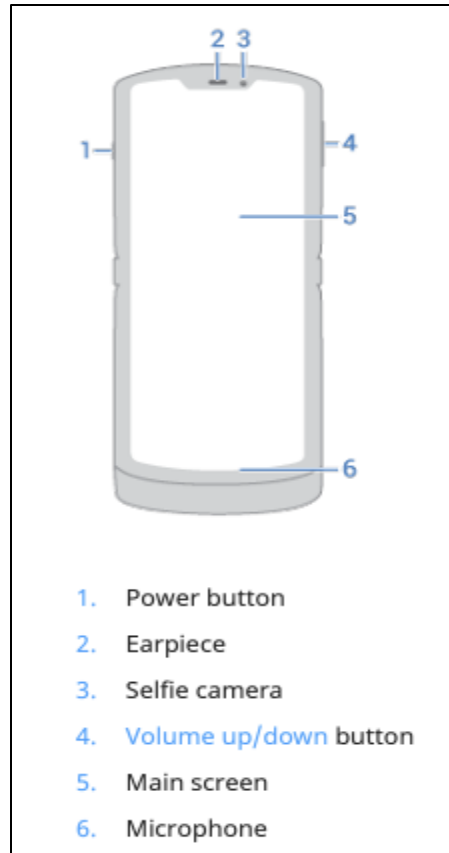
65. Paragraphs 1 through 51 are incorporated by reference as if fully set forth herein.

66. Plaintiffs have not licensed or otherwise authorized Defendants to make, use, offer for sale, sell, or import any products that embody the inventions of the '930 Patent.

67. Defendants have and continue to directly infringe the '930 Patent, either literally or under the doctrine of equivalents, without authority and in violation of 35 U.S.C. § 271, by making, using, testing, offering to sell, selling, and/or importing into the United States products that satisfy each and every limitation of one or more claims of the '930 Patent. Such products include Lenovo smartphones, such as the Motorola Razr, having a secure authentication and/or biometric sensor functionality, such as a fingerprint sensor or face unlock feature.

68. For example, Defendants have and continue to directly infringe at least claim 10 of the '930 Patent by making, using, testing, offering to sell, selling, and/or importing into the United States products including, but not limited to, Lenovo smartphones, such as the Motorola Razr, having a secure authentication and/or biometric sensor functionality, such as a fingerprint sensor or face unlock feature.

69. The Motorola Razr performs a method for providing and using trusted user input on a computing platform. For example, the Motorola Razr can use authentication (e.g., a pin) or biometric sensing (e.g., a fingerprint or face scanner) to permit certain functions, such as mobile payments.



Screen lock types

No lock:

- **None:** This gives no protection, but you can quickly access your home screen.
- **Swipe:** Swipe  up. This prevents accidental dialing but doesn't secure the phone.

Lock:

- **Pattern:** Draw a pattern on a grid.
- **PIN:** Enter a four to sixteen digit numeric code. Longer PINs are more secure.
- **Password:** Enter four to sixteen letters, digits, or symbols for the password. A long, strong password is the most secure option.

When you're using any of the 3 locks, you can also [unlock with your fingerprint](#) and set your phone to [automatically unlock](#) in certain conditions.

Unlock your screen automatically

If you're using a PIN, password, or pattern, you can:

- Stay unlocked when [connected to a device](#), like your car audio system.
- Stay unlocked at a [specific location](#).
- Stay unlocked when [holding or carrying](#) your phone.
- Unlock with [your face](#).

Tap & pay with NFC

Only some versions of this phone, sold in certain countries, support this feature.

Use NFC (near-field communication) to pay quickly and [securely](#) at checkout registers by touching your phone to an electronic reader.

Set up Tap & pay

1. Make sure you have:
 - Set up a payment app on your phone. [Install one](#) from Play Store if needed.
 - Turned on [NFC](#).
2. Go to [Settings](#) > **Apps & notifications**.
3. Touch **Advanced** > **Special app access** > **Tap & pay**.
4. Touch **Payment default**.
5. Touch an app to make it your default payment app.

Note: This feature manages payment apps. It doesn't include money transfer apps.

Make a payment

1. Make sure [NFC is on](#) and that the terminal has this symbol:



2. Unlock your phone and touch its [back](#) to the terminal.

Your payment app might require you to open the app and enter a PIN before touching your phone to the terminal.

Turn off Tap & pay

[Turn off NFC](#).

When NFC is off, Tap & pay is not included in [Settings](#).

27

²⁷ Motorola Razr User Guide, available at: <https://help.motorola.com/hc/5502/10/pdf/help-razr-10-esm-en-us.pdf>

AOSP > Secure > Features
Rate and review  

Android Security Features

Use the features described in this section to make the Android devices you develop as secure as possible.

Authentication

Android uses the concept of user-authentication-gated cryptographic keys that requires cryptographic key storage and service provider and user authenticators.

On devices with a fingerprint sensor, users can enroll one or more fingerprints and use those fingerprints to unlock the device and perform other tasks. The Gatekeeper subsystem performs device pattern/password authentication in a Trusted Execution Environment (TEE).

Android 9 and higher includes Protected Confirmation, which gives users a way to formally confirm critical transactions, such as payments.

Biometrics

Android 9 and higher includes a BiometricPrompt API that app developers can use to integrate biometric authentication into their apps in a device- and modality-agnostic fashion. Only strong biometrics can integrate with `BiometricPrompt`.

Trusty Trusted Execution Environment (TEE)

Trusty is a secure Operating System (OS) that provides a Trusted Execution Environment (TEE) for Android. The Trusty OS runs on the same processor as the Android OS, but Trusty is isolated from the rest of the system by both hardware and software.

28

70. The Motorola Razr performs the step of instantiating in the computing platform an isolated operating environment for user information (e.g., the Trusty Trusted Execution Environment) adapted for secure processing of information relating to a user. For example, the Motorola Razr allows for authentic and biometric information to be stored in secure environment, such as Trusted Execution Environment (“TEE”), a Secure Element (“SE”), such as Strongbox, or Gatekeeper.

²⁸ <https://source.android.com/security/features>

AOSP > Secure > Features

Rate and review  

Android Security Features

Use the features described in this section to make the Android devices you develop as secure as possible.

Authentication

Android uses the concept of user-authentication-gated cryptographic keys that requires cryptographic key storage and service provider and user authenticators.

On devices with a fingerprint sensor, users can enroll one or more fingerprints and use those fingerprints to unlock the device and perform other tasks. The Gatekeeper subsystem performs device pattern/password authentication in a Trusted Execution Environment (TEE).

Android 9 and higher includes Protected Confirmation, which gives users a way to formally confirm critical transactions, such as payments.

Biometrics

Android 9 and higher includes a BiometricPrompt API that app developers can use to integrate biometric authentication into their apps in a device- and modality-agnostic fashion. Only strong biometrics can integrate with `BiometricPrompt`.

Trusty Trusted Execution Environment (TEE)

Trusty is a secure Operating System (OS) that provides a Trusted Execution Environment (TEE) for Android. The Trusty OS runs on the same processor as the Android OS, but Trusty is isolated from the rest of the system by both hardware and software.

29

²⁹ <https://source.android.com/security/features>

Authentication

Android uses the concept of user-authentication-gated cryptographic keys that requires the following components:

- **Cryptographic key storage and service provider.** Stores cryptographic keys and provides standard crypto routines on top of those keys. Android supports a [hardware-backed Keystore](#) and Keymaster for cryptographic services, including hardware-backed cryptography for key storage that might include a Trusted Execution Environment (TEE) or Secure Element (SE), such as Strongbox.
- **User authenticators.** Attest to the user's presence and/or successful authentication. Android supports [Gatekeeper](#) for PIN/pattern/password authentication and [Fingerprint](#) for fingerprint authentication. Devices that ship with Android 9 and higher can use [BiometricPrompt](#) [↗](#) as a single integration point for fingerprint and additional biometrics. These components communicate their authentication state with the keystore service through an authenticated channel. (The [Android Keystore system](#) [↗](#) at the framework level is also backed by the keystore service.)

Gatekeeper, Fingerprint, and Biometric components work with Keystore and other components to support the use of hardware-backed [authentication tokens](#) (AuthTokens).

Enrollment

On first boot of the device after a factory reset, all authenticators are prepared to receive credential enrollments from the user. A user must initially enroll a PIN/pattern/password with Gatekeeper. This initial enrollment creates a randomly generated, 64-bit user secure identifier (SID) that serves as an identifier for the user and as a binding token for the user's cryptographic material. This user SID is cryptographically bound to the user's password; successful authentications to Gatekeeper result in AuthTokens that contain the user SID for that password.

A user who wants to change a credential must present an existing credential. If an existing credential is verified successfully, the user SID associated with the existing credential is transferred to the new credential, enabling the user to keep accessing keys after changing a credential. If a user does not present an existing credential, the new credential is enrolled with a fully random User SID. The user can access the device, but keys created under the old user SID are permanently lost. This is known as an *untrusted enroll*.

Under normal circumstances, the Android framework doesn't allow an untrusted enroll, so most users won't ever see this functionality. However, forcible password resets, either by a device administrator or an attacker, may cause this to occur.

Authentication

After a user has set up a credential and received a user SID, they can start authentication, which begins when a user provides a PIN, pattern, password, or fingerprint. All TEE components share a secret key that they use to authenticate each other's messages.

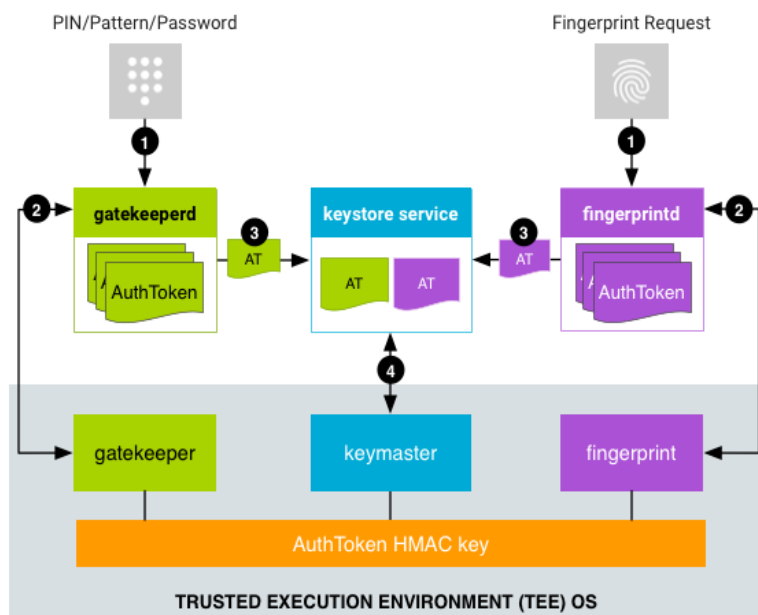


Figure 1. Authentication flow

30

71. The Motorola Razr performs the step of the isolated operating environment for user information (e.g., a TEE) receiving a request for a service relating to use of trusted user input.

³⁰ <https://source.android.com/security/authentication>

1. A user provides an authentication method and the associated service makes a request to the associated daemon.
 - For PIN, pattern, or password, `LockSettingsService` makes a request to `gatekeeperd`.
 - Biometrics-based authentication flows depend on the Android version. On devices running Android 8.x and lower, `FingerprintService` makes a request to `fingerprintd`. On devices running Android 9 and higher, `BiometricPrompt` makes a request to the appropriate biometric daemon (for example, `fingerprintd` for fingerprints or `faced` for face) using the appropriate `BiometricManager` class, such as `FingerprintManager` or `FaceManager`. Regardless of version, biometric authentication occurs asynchronously after the request is sent.
2. The daemon sends data to its counterpart, which generates an AuthToken:
 - For PIN/pattern/password authentication, `gatekeeperd` sends the PIN, pattern, or password hash to Gatekeeper in the TEE. If authentication in the TEE is successful, Gatekeeper in the TEE sends an AuthToken containing the corresponding user SID (signed with the AuthToken HMAC key) to its counterpart in the Android OS.
 - For fingerprint authentication, `fingerprintd` listens for fingerprint events and sends the data to Fingerprint in the TEE. If authentication in the TEE is successful, Fingerprint in the TEE sends an AuthToken (signed with the AuthToken HMAC key) to its counterpart in the Android OS.
 - For other biometric authentication, the appropriate biometric daemon listens for the biometric event and sends it to the appropriate biometric TEE component.
3. The daemon receives a signed AuthToken and passes it to the keystore service through an extension to the keystore service's Binder interface. (`gatekeeperd` also notifies the keystore service when the device is relocated and when the device password changes.)
4. The keystore service passes the AuthTokens to Keymaster and verifies them using the key shared with the Gatekeeper and supported biometric TEE component. Keymaster trusts the timestamp in the token as the last authentication time and bases a key release decision (to allow an app to use the key) on the timestamp.

31

7.3.10. Biometric Sensors

For additional background on Measuring Biometric Unlock Security, please see [Measuring Biometric Security documentation](#).

If device implementations include a secure lock screen, they:

- SHOULD include a biometric sensor

Biometric sensors can be classified as Class 3 (formerly Strong), Class 2 (formerly Weak), or Class 1 (formerly Convenience) based on their spoof and imposter acceptance rates, and on the security of the biometric pipeline. This classification determines the capabilities the biometric sensor has to interface with the platform and with third-party applications. Sensors are classified as Class 1 by default, and need to meet additional requirements as detailed below if they wish to be classified as either Class 2 or Class 3. Both Class 2 and Class 3 biometrics get additional capabilities as detailed below.

If device implementations make a biometric sensor available to third-party applications via

³¹ <https://source.android.com/security/authentication>

`android.hardware.biometrics.BiometricManager`, `android.hardware.biometrics.BiometricPrompt`, and `android.provider.Settings.ACTION_BIOMETRIC_ENROLL`, they:

- [C-4-1] MUST meet the requirements for Class 3 or Class 2 **biometric** as defined in this document.
- [C-4-2] MUST recognize and honor each parameter name defined as a constant in the `Authenticators` class and any combinations thereof. Conversely, MUST NOT honor or recognize integer constants passed to the `canAuthenticate(int)` and `setAllowedAuthenticators(int)` methods other than those documented as public constants in `Authenticators` and any combinations thereof.
- [C-4-3] MUST implement the `ACTION_BIOMETRIC_ENROLL` action on devices that have either Class 3 or Class 2 biometrics. This action MUST only present the enrollment entry points for Class 3 or Class 2 biometrics.

If device implementations wish to treat a biometric sensor as Class 3 (formerly Strong), they:

- [C-3-1] MUST meet all the requirements of Class 2 above, except for [C-1-7] and [C-1-8]. Upgrading devices from an earlier Android version are not exempted from C-2-7.
- [C-3-2] MUST have a hardware-backed keystore implementation.
- [C-3-3] MUST have a spoof and imposter acceptance rate not higher than 7% as measured by the *Android Biometrics Test Protocols*.
- [C-3-4] MUST challenge the user for the recommended primary authentication (e.g. PIN, pattern, password) once every 72 hours or less.

32

72. The Motorola Razr performs the step of the isolated operating environment for user information (e.g., TEE) establishing that trusted user input (e.g., a pin, fingerprint scan, or face scan) can safely be received from an isolated operating environment for input adapted to receive reliably and retain securely user input as provided by a user into a user input device.

³² <https://source.android.com/compatibility/11/android-11-cdd.pdf>

If device implementations wish to treat a biometric sensor as Class 2 (formerly Weak), they:

- [C-2-1] MUST meet all requirements for Class 1 above.
- [C-2-2] MUST have a spoof and imposter acceptance rate not higher than 20% as measured by the [Android Biometrics Test Protocols](#) .
- [C-2-3] MUST perform the biometric matching in an isolated execution environment outside Android user or kernel space, such as the Trusted Execution Environment (TEE), or on a chip with a secure channel to the isolated execution environment.
- [C-2-4] MUST have all identifiable data encrypted and cryptographically authenticated such that they cannot be acquired, read or altered outside of the isolated execution environment or a chip with a secure channel to the isolated execution environment as documented in the [implementation guidelines](#) on the Android Open Source Project site.
- [C-2-5] For camera based biometrics, while biometric based authentication or enrollment is happening:
 - MUST operate the camera in a mode that prevents camera frames from being read or altered outside the isolated execution environment or a chip with a secure channel to the isolated execution environment.
 - For RGB single-camera solutions, the camera frames CAN be readable outside the isolated execution environment to support operations such as preview for enrollment, but MUST still NOT be alterable.
- [C-2-6] MUST NOT enable third-party applications to distinguish between individual biometric enrollments.
- [C-2-7] MUST NOT allow unencrypted access to identifiable biometric data or any data derived from it (such as embeddings) to the Application Processor outside the context of the TEE.
- [C-2-8] MUST have a secure processing pipeline such that an operating system or kernel compromise cannot allow data to be directly injected to falsely authenticate as the user.
If device implementations are already launched on an earlier Android version and cannot meet the requirement C-2-8 through a system software update, they MAY be exempted from the requirement.
- [C-SR] Are STRONGLY RECOMMENDED to include liveness detection for all biometric modalities and attention detection for Face biometrics.

33

³³ <https://source.android.com/compatibility/11/android-11-cdd.pdf>

HAL implementation guidelines

Follow these biometric HAL guidelines to ensure that biometric data is **not leaked** and is **removed** when a user is removed from a device:

- Make sure that raw biometric data or derivatives (such as templates) are never accessible from outside the secure isolated environment (such as the TEE or Secure Element). All stored data must be encrypted with a **device-specific** key known only to the TEE (Trusted Execution Environment). If the hardware supports it, limit hardware access to the secure isolated environment and protect it with an SELinux policy. Make the communication channel (for example, SPI, I2C) accessible only to the secure isolated environment with an explicit SELinux policy on all device files.
- Biometric acquisition, enrollment, and recognition must occur inside the secure isolated environment to prevent data breaches and other attacks. This requirement only applies to **Class 3 (formerly Strong) and Class 2 (formerly Weak)** biometrics.
- To protect against replay attacks, sign biometric templates with a private, device-specific key. For Advanced Encryption Standard (AES), at a minimum sign a template with the absolute file-system path, group, and biometric ID such that template files are inoperable on another device or for anyone other than the user that enrolled them on the same device. For example, prevent copying biometric data from a different user on the same device or from another device.
- If you need to store data outside of the TEE, use the file-system path provided by the `setActiveUser()` `HIDL method` or provide another way to erase all user template data when the user is removed. The reason is to protect leaking of user data. Devices that don't use this path **must** clean up after the user is removed. It's required by CDD that biometric data and derivative files be stored encrypted - especially if not in TEE. If this is infeasible due to the storage requirements of the secure isolated environment, add hooks to ensure removal of the data when the user is removed or the device is wiped. See `LockSettingsService.removeBiometricsForUser()`

34

73. The Motorola Razr performs the step of the isolated operating environment for user information (e.g., a TEE) receiving the trusted user input securely from the isolated operating environment for input along a trusted path therebetween; and the isolated operating environment for user information using the trusted user input to provide the requested service relating to use of the trusted user input.

³⁴ <https://source.android.com/security/biometric#hal-implementation>

If device implementations wish to treat a biometric sensor as Class 2 (formerly Weak), they:

- [C-2-1] MUST meet all requirements for Class 1 above.
- [C-2-2] MUST have a spoof and imposter acceptance rate not higher than 20% as measured by the [Android Biometrics Test Protocols](#) .
- [C-2-3] MUST perform the biometric matching in an isolated execution environment outside Android user or kernel space, such as the Trusted Execution Environment (TEE), or on a chip with a secure channel to the isolated execution environment.
- [C-2-4] MUST have all identifiable data encrypted and cryptographically authenticated such that they cannot be acquired, read or altered outside of the isolated execution environment or a chip with a secure channel to the isolated execution environment as documented in the [implementation guidelines](#) on the Android Open Source Project site.
- [C-2-5] For camera based biometrics, while biometric based authentication or enrollment is happening:
 - MUST operate the camera in a mode that prevents camera frames from being read or altered outside the isolated execution environment or a chip with a secure channel to the isolated execution environment.
 - For RGB single-camera solutions, the camera frames CAN be readable outside the isolated execution environment to support operations such as preview for enrollment, but MUST still NOT be alterable.
- [C-2-6] MUST NOT enable third-party applications to distinguish between individual biometric enrollments.
- [C-2-7] MUST NOT allow unencrypted access to identifiable biometric data or any data derived from it (such as embeddings) to the Application Processor outside the context of the TEE.
- [C-2-8] MUST have a secure processing pipeline such that an operating system or kernel compromise cannot allow data to be directly injected to falsely authenticate as the user. If device implementations are already launched on an earlier Android version and cannot meet the requirement C-2-8 through a system software update, they MAY be exempted from the requirement.
- [C-SR] Are STRONGLY RECOMMENDED to include liveness detection for all biometric modalities and attention detection for Face biometrics.

³⁵ <https://source.android.com/compatibility/11/android-11-cdd.pdf>

HAL implementation guidelines

Follow these biometric HAL guidelines to ensure that biometric data is **not leaked** and is **removed** when a user is removed from a device:

- Make sure that raw biometric data or derivatives (such as templates) are never accessible from outside the secure isolated environment (such as the TEE or Secure Element). All stored data must be encrypted with a **device-specific** key known only to the TEE (Trusted Execution Environment). If the hardware supports it, limit hardware access to the secure isolated environment and protect it with an SELinux policy. Make the communication channel (for example, SPI, I2C) accessible only to the secure isolated environment with an explicit SELinux policy on all device files.
- Biometric acquisition, enrollment, and recognition must occur inside the secure isolated environment to prevent data breaches and other attacks. This requirement only applies to **Class 3 (formerly Strong) and Class 2 (formerly Weak)** biometrics.
- To protect against replay attacks, sign biometric templates with a private, device-specific key. For Advanced Encryption Standard (AES), at a minimum sign a template with the absolute file-system path, group, and biometric ID such that template files are inoperable on another device or for anyone other than the user that enrolled them on the same device. For example, prevent copying biometric data from a different user on the same device or from another device.
- If you need to store data outside of the TEE, use the file-system path provided by the `setActiveUser()` [HIDL method](#) or provide another way to erase all user template data when the user is removed. The reason is to protect leaking of user data. Devices that don't use this path **must** clean up after the user is removed. It's required by CDD that biometric data and derivative files be stored encrypted - especially if not in TEE. If this is infeasible due to the storage requirements of the secure isolated environment, add hooks to ensure removal of the data when the user is removed or the device is wiped. See `LockSettingsService.removeBiometricsForUser()`

36

74. Lenovo directly infringes the '930 Patent through promotional demonstrations, testing, repairs, and instructional guidance of the Motorola Razr, performing at least the steps of claim 10.

75. Defendants have and continue to indirectly infringe one or more claims of the '930 Patent by knowingly and intentionally inducing others, including Lenovo customers and end-users, to directly infringe, either literally or under the doctrine of equivalents, by making, using, offering to sell, selling, and/or importing into the United States products that include the infringing technology.

³⁶ <https://source.android.com/security/biometric#hal-implementation>

76. Defendants, with knowledge³⁷ that these products, or the use thereof, infringe the '930 Patent at least as early as February 19, 2022³⁸, knowingly and intentionally induced, and continue to knowingly and intentionally induce, direct infringement of the '930 Patent by providing these products to end-users for use in an infringing manner. Additionally, on information and belief, Defendants have adopted a policy of not reviewing the patents of others, including specifically those related to Defendants' specific industry, thereby remaining willfully blind to the '930 Patent at least as early as the issuance of the '930 Patent.

77. Defendants have induced infringement by others, including end-users, with the intent to cause infringing acts by others or, in the alternative, with the belief that there was a high probability that others, including end-users, infringe the '930 Patent, but while remaining willfully blind to the infringement. Defendants have and continue to induce infringement by their customers and end-users by supplying them with instructions on how to operate the infringing technology in an infringing manner, while also making publicly available information on the infringing technology via Defendants' website, product literature and packaging, and other publications.³⁹

78. Plaintiffs have suffered damages as a result of Defendants' direct and indirect infringement of the '930 Patent in an amount to be proven at trial.

79. Plaintiffs have suffered, and will continue to suffer, irreparable harm as a result of

³⁷ Lenovo cited directly to the '930 Patent family against its own U.S. Patent Application No. 11/865,048, which was published on April 2, 2009, with Publication No. U.S. 2009/0089875 A1, and issued as U.S. Patent No. 8,205,248.

³⁸ During the negotiations between Plaintiffs and Defendant Lenovo Group Limited, Plaintiffs sent an email to Defendant Lenovo Group Limited on February 19, 2022, notifying them of its infringement of the '930 Patent. Plaintiffs continued negotiating a license to the '930 Patent, among others, until at least November 2024. Therefore, at least Defendant Lenovo Group Limited has been on notice of its infringement of the '930 Patent since no later than February 19, 2022.

³⁹ See, e.g., Motorola Razr User Guide (HTML), available at: [https://en-us.support.motorola.com/app/answers/detail/a_id/147758/~/-motorola-razr---user-guide-\(html\)](https://en-us.support.motorola.com/app/answers/detail/a_id/147758/~/-motorola-razr---user-guide-(html))

Defendants' infringement of the '930 Patent, for which there is no adequate remedy at law, unless Defendants' infringement is enjoined by this Court.

80. Defendants have committed and continue to commit acts of infringement that Defendants actually knew or should have known constituted an unjustifiably high risk of infringement of at least one valid and enforceable claim of the '930 Patent. Defendants' direct and indirect infringement of the '930 Patent has been and continues to be willful, intentional, deliberate, and/or in conscious disregard of rights under the patent. Plaintiffs are entitled to an award of treble damages, reasonable attorney fees, and costs in bringing this action.

COUNT III
(Infringement of the '539 Patent)

81. Paragraphs 1 through 51 are incorporated by reference as if fully set forth herein.

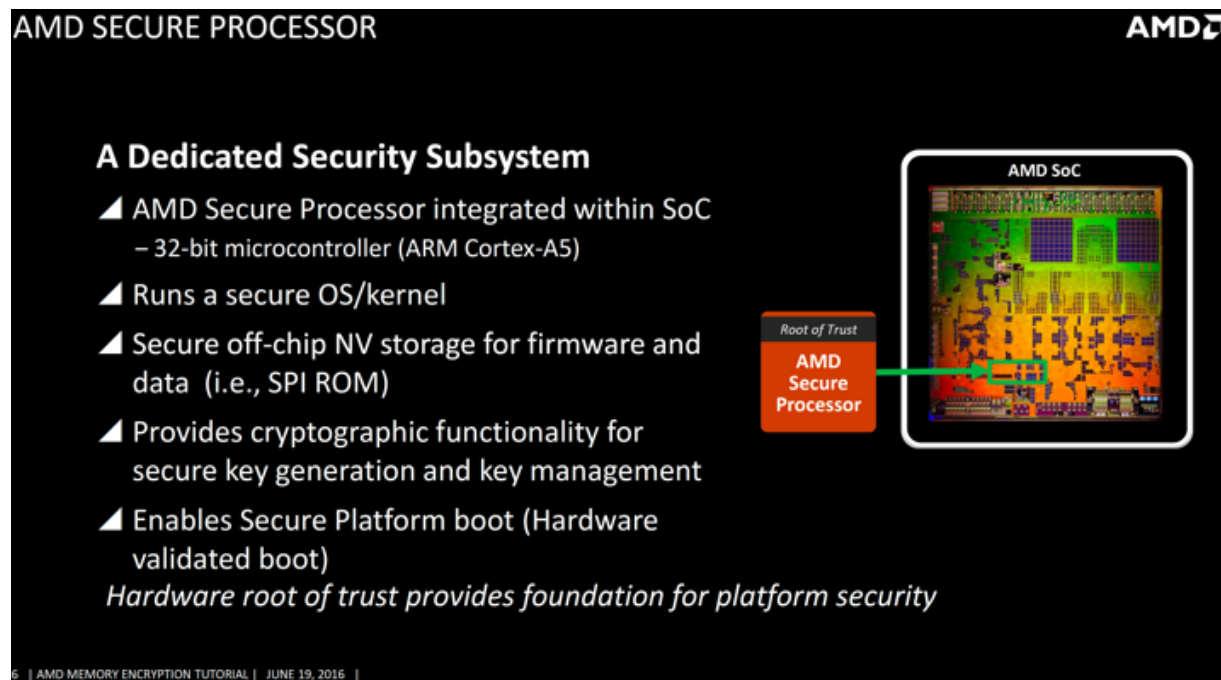
82. Plaintiffs have not licensed or otherwise authorized Defendants to make, use, offer for sale, sell, or import any products that embody the inventions of the '539 Patent.

83. Defendants have and continue to directly infringe the '539 Patent, either literally or under the doctrine of equivalents, without authority and in violation of 35 U.S.C. § 271, by making, using, offering to sell, selling, and/or importing into the United States products that satisfy each and every limitation of one or more claims of the '539 Patent. Such products include Lenovo devices equipped with AMD multicore processors, AMD EPYC processors, AMD Ryzen processors, and AMD Athlon processors, such as Lenovo servers, desktops, laptops, and workstations, such as the Lenovo ThinkSystem SR665.

84. For example, Defendants have and continue to directly infringe at least claim 1 of the '539 Patent by making, using, offering to sell, selling, and/or importing into the United States products that include Lenovo devices equipped with AMD multicore processors, AMD EPYC processors, AMD Ryzen processors, and AMD Athlon processors, such as Lenovo servers,

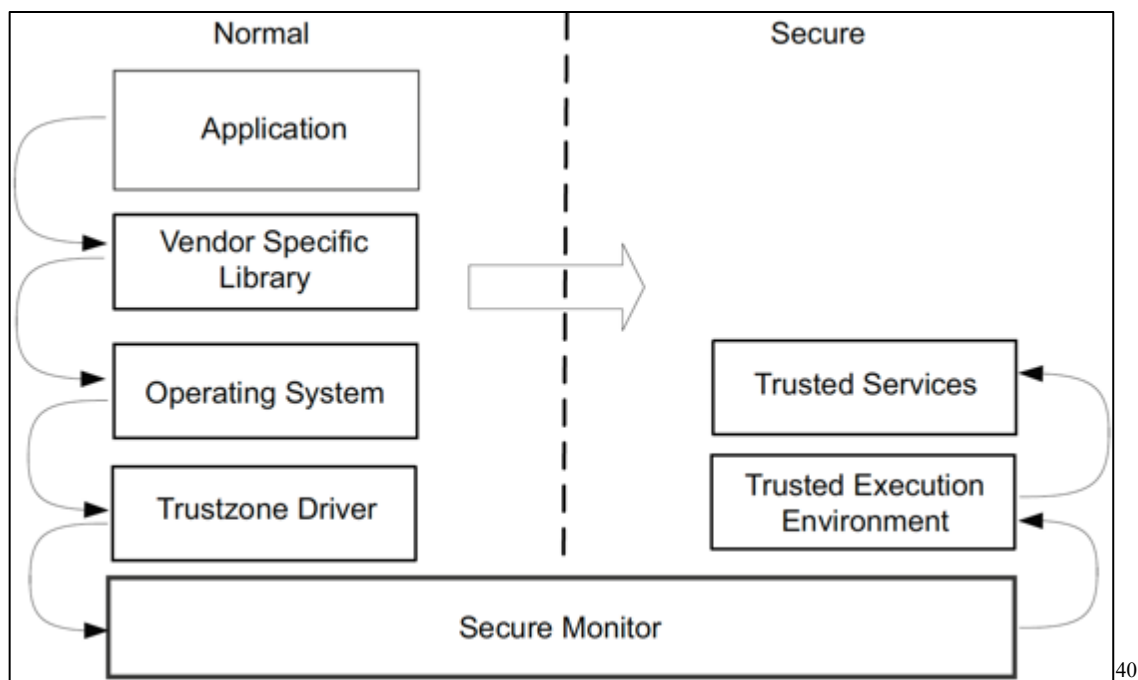
desktops, laptops, and workstations, such as the Lenovo ThinkSystem SR665.

85. The Lenovo ThinkSystem SR665, equipped with the AMD EPYC 7003 Series Processors, performs a computer implemented method for use in a computer system including a plurality of resources. For example, the method is executed by a device featuring a processor where the device includes resources, such as normal (non-secure) code and secure code.



86. The Lenovo ThinkSystem SR665, equipped with the AMD EPYC 7003 Series Processors, performs the step of receiving a request from a software program to access a specified one of the plurality of resources. For example, a software program, such as the TrustZone API driver in the normal (non-secure) world, requests to access a resource, such as a normal (non-secure) or secure application code, on behalf of an application in the normal (non-secure) world.

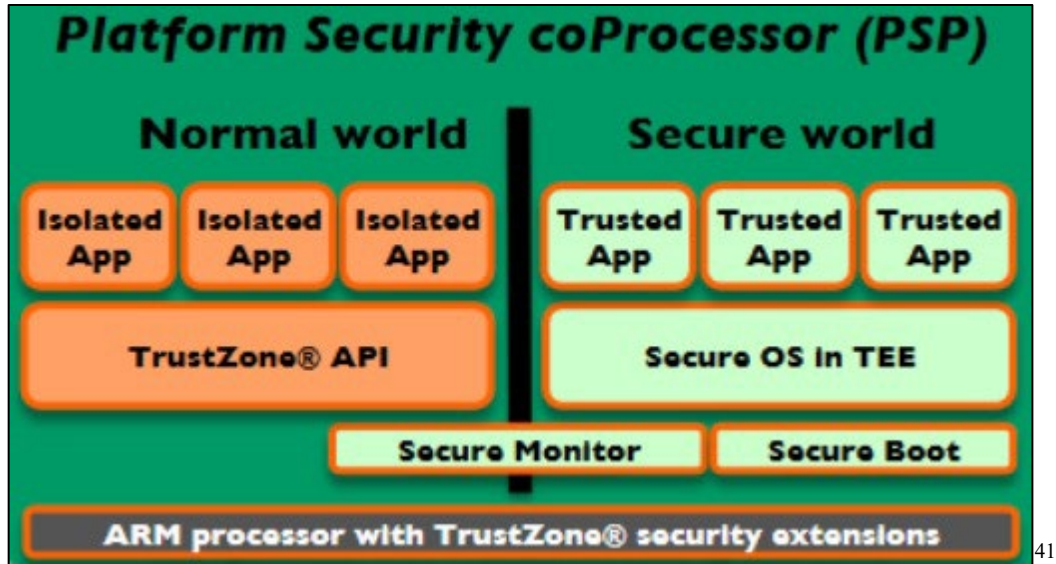
Generally applications developers won't directly interact with TrustZone (or TEEs or Trusted Services). Instead, one makes use of a high level API (for example, it might be called `reqPayment()`) provided by a Normal world library. The library would be provided by the same vendor as the Trusted Service (for example, a credit card company), and would handle the low level interactions. Figure 21-3 shows this interaction and illustrates the flow from user application calling the API that makes an appropriate OS call, which then passes to the TrustZone driver code, and then passes execution into the TEE through the Secure monitor.



87. The Lenovo ThinkSystem SR665, equipped with the AMD EPYC 7003 Series Processors, performs the step of determining whether the specified one of the plurality of resources is a protected resource. For example, the resource is determined if it is a protected resource, such as a secure application code performing security service in the secure world. The AMD Secure Processor (AMD-SP) is known as a Platform Security Coprocessor (PSP).

If you do have to access a secure application, you will require a driver-like function to talk to the Secure world OS and Secure applications, but the details of creating that Secure world OS

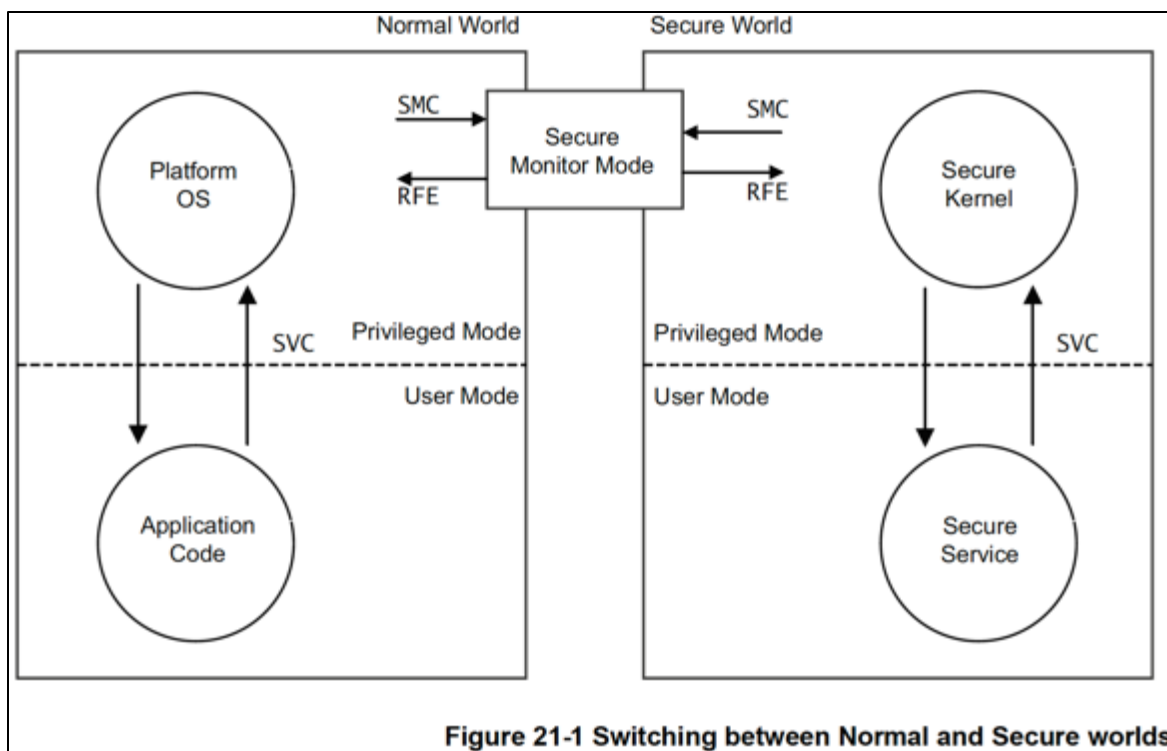
⁴⁰ <https://developer.arm.com/documentation/den0013/d/Security/TrustZone-hardware-architecture/Interaction-of-Normal-and-Secure-worlds>



88. The Lenovo ThinkSystem SR665, equipped with the AMD EPYC 7003 Series Processors, performs the step of, if the specified one of the plurality of resources is a protected resource. For example, if the resource is a protected resource – a resource application code performing security service in the secure world – then a secure monitor call (SMC) is made to the secure monitor in order for the secure code to be executed in the secure world.

It is common to share data between the Secure and Normal worlds. For example, in the Secure world you might have a signature checker. The Normal world can request that the Secure world verifies the signature of a downloaded update, using the SMC call. The Secure world requires

⁴¹ <https://freundschafter.com/research/about-amd-trustzone-amd-platform-security-processor-psp-amd-secure-technology/>



The Secure Monitor Call instruction, SMC, requests a Secure Monitor function, causing the processor to enter Monitor mode. For more information, see *SMC (previously SMI)* on page B9-1988.

42

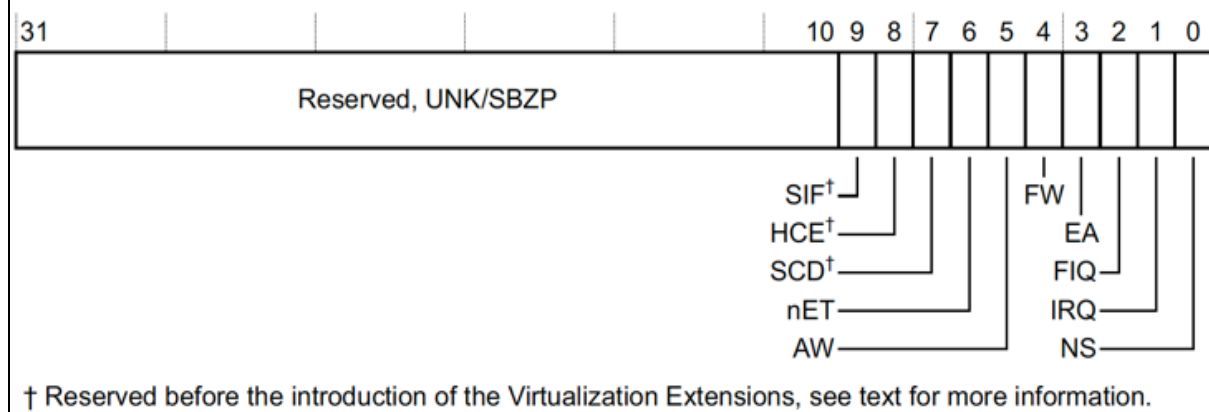
89. The Lenovo ThinkSystem SR665, equipped with the AMD EPYC 7003 Series Processors, performs the step of, if the computer system is operating in a protected mode of operation, then denying the request regardless of access rights associated with the software program including software programs having a most-privileged level. For example, if the computer system (processor core) is operating in a protected mode of operation, such as defined by setting SCR_SCD to 1 and hence disabling the secure monitor call at PL1 and above, then the request is denied regardless of access rights and the privileged level, since SMC cannot execute and the system cannot switch to the secure world to access the secure application code.

⁴² <https://developer.arm.com/documentation/den0013/d/Security/TrustZone-hardware-architecture/Interaction-of-Normal-and-Secure-worlds>

When the `SCR.SCD` bit is set to 1, entry to Secure state by taking a Secure Monitor Call exception is disabled. This means that, when `SCR.SCD` is set to 1:

- An SMC instruction executed in Non-secure state, and not trapped by the HCR.TSC mechanism described in *Trapping use of the SMC instruction on page B1-1253*, is UNDEFINED.
- An SMC instruction executed in a Secure PL1 mode is UNPREDICTABLE.

The SCR bit assignments are:



SCD, bit[7], when implementation includes the Virtualization Extensions

Secure Monitor Call disable. Makes the SMC instruction UNDEFINED in Non-secure state. The possible values of this bit are:

- | | |
|---|--|
| 0 | SMC executes normally in Non-secure state, performing a Secure Monitor Call. |
| 1 | SMC instruction is UNDEFINED in Non-secure state. |

43

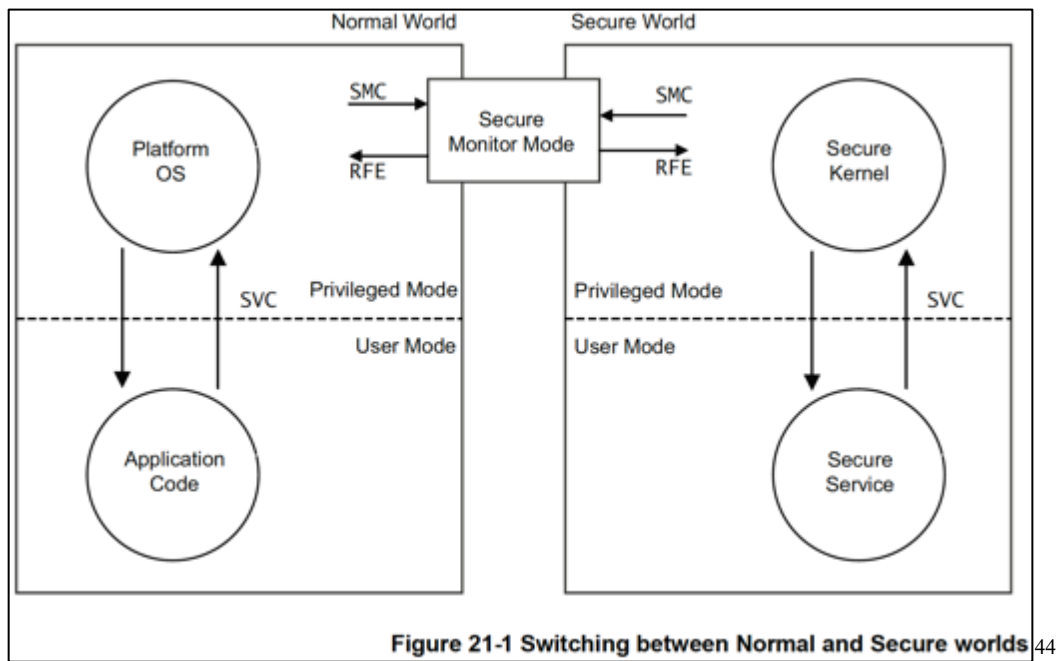
90. The Lenovo ThinkSystem SR665, equipped with the AMD EPYC 7003 Series Processors, performs the step of processing the request based on the access rights associated with the software program if the computer system is not operating in the protected mode of operation. For example, if the computer system (e.g., processor core) is not operating in a protected mode of operation, such as defined by setting SCR.SMD to 0 and hence enabling the secure monitor call, then the request is processed based on the access rights, such as to SMC, associated with the software program.

⁴³ <https://developer.arm.com/documentation/ddi0406/cd>

It is common to share data between the Secure and Normal worlds. For example, in the Secure world you might have a signature checker. The Normal world can request that the Secure world verifies the signature of a downloaded update, using the SMC call. The Secure world requires

Secure Monitor Call causes a Secure Monitor Call exception. For more information, see *Secure Monitor Call (SMC) exception* on page B1-1210.

SMC is available only from software executing at PL1 or higher. It is UNDEFINED in User mode.



91. Lenovo directly infringes the '539 Patent through promotional demonstrations, testing, repairs, and instructional guidance of the Lenovo ThinkSystem SR665, performing at least the steps of claim 1.

92. Defendants have and continue to indirectly infringe one or more claims of the '539 Patent by knowingly and intentionally inducing others, including Lenovo customers and end-users, to directly infringe, either literally or under the doctrine of equivalents, by making, using, offering to sell, selling, and/or importing into the United States products that include the infringing technology.

⁴⁴<https://developer.arm.com/documentation/den0013/d/Security/TrustZone-hardware-architecture/Interaction-of-Normal-and-Secure-worlds>

93. Defendants, with knowledge⁴⁵ that these products, or the use thereof, infringe the '539 Patent at least as of September 16, 2022,⁴⁶ knowingly and intentionally induced, and continue to knowingly and intentionally induce, direct infringement of the '539 Patent by providing these products to end-users for use in an infringing manner. Additionally, on information and belief, Defendants have adopted a policy of not reviewing the patents of others, including specifically those related to Defendants' specific industry, thereby remaining willfully blind to the '539 Patent at least as early as the issuance of the '539 Patent.

94. Defendants have induced infringement by others, including end-users, with the intent to cause infringing acts by others or, in the alternative, with the belief that there was a high probability that others, including end-users, infringe the '539 Patent, but while remaining willfully blind to the infringement. Defendants have and continue to induce infringement by their customers and end-users by supplying them with instructions on how to operate the infringing technology in an infringing manner, while also making publicly available information on the infringing technology via Defendants' website, product literature and packaging, and other publications.⁴⁷

95. Plaintiffs have suffered damages as a result of Defendants' direct and indirect infringement of the '539 Patent in an amount to be proven at trial.

⁴⁵ Lenovo cited to the '539 Patent family against its own U.S. Patent Application 11/767,266, which was published on January 24, 2008, with Publication No. U.S. 2008/0022376 A1.

⁴⁶ During the negotiations between Plaintiffs and Defendant Lenovo Group Limited, Plaintiffs sent an email to Defendant Lenovo Group Limited on September 16, 2022, notifying them of its infringement of the '539 Patent. Plaintiffs continued negotiating a license to the '539 Patent, among others, until at least November 2024. Therefore, at least Defendant Lenovo Group Limited has been on notice of its infringement of the '539 Patent since at least September 16, 2022.

⁴⁷ See, e.g., Lenovo ThinkSystem SR665 Setup Guide, available at: https://pubs.lenovo.com/sr665/sr665_setup_guide.pdf

96. Plaintiffs have suffered, and will continue to suffer, irreparable harm as a result of Defendants' infringement of the '539 Patent, for which there is no adequate remedy at law, unless Defendants' infringement is enjoined by this Court.

97. Defendants have committed and continue to commit acts of infringement that Defendants actually knew or should have known constituted an unjustifiably high risk of infringement of at least one valid and enforceable claim of the '539 Patent. Defendants' direct and indirect infringement of the '539 Patent has been and continues to be willful, intentional, deliberate, and/or in conscious disregard of rights under the patent. Plaintiffs are entitled to an award of treble damages, reasonable attorney fees, and costs in bringing this action.

COUNT IV
(Infringement of the '332 Patent)

98. Paragraphs 1 through 51 are incorporated by reference as if fully set forth herein.

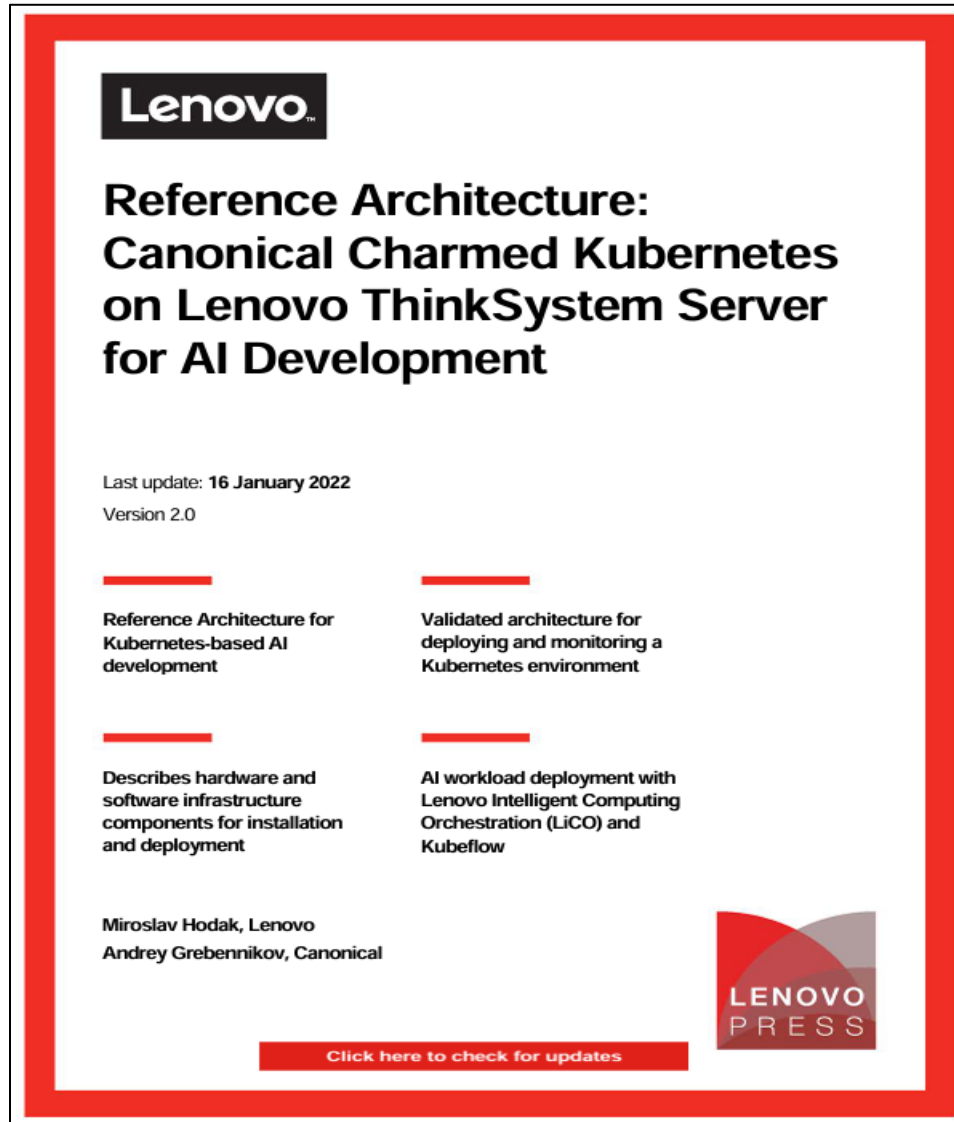
99. Plaintiffs have not licensed or otherwise authorized Defendants to make, use, offer for sale, sell, or import any products that embody the inventions of the '332 Patent.

100. Defendants have and continue to directly infringe the '332 Patent, either literally or under the doctrine of equivalents, without authority and in violation of 35 U.S.C. § 271, by making, using, offering to sell, selling, and/or importing into the United States products that satisfy each and every limitation of one or more claims of the '332 Patent. Such products include Lenovo smartphones or computers, and/or servers. The technology is implemented in systems using Kubernetes for rolling updates, such as those offered by Lenovo's Infrastructure Solutions Group including, but not limited to, ThinkSystem and ThinkAgile servers.

101. For example, Defendants have and continue to directly infringe at least claim 1 of the '332 Patent by making, using, offering to sell, selling, and/or importing into the United States products that include systems using Kubernetes for rolling updates, such as the Lenovo

ThinkSystem SR665, among other Lenovo products.

102. Upon information and belief, Lenovo ThinkSystem servers implement or otherwise utilize Kubernetes.



1 Introduction

This document describes the Lenovo hardware specifications as well as the tools and services to deploy the solution, including the foundation cluster, the Kubernetes cluster and tools used for cluster monitoring and management.

This guide also provides the deployment steps and references to configuration and automation scripts developed by Lenovo and Canonical for the deployment process. Examples for validating the deployed solution and the expected results are provided as well to ensure a successful implementation of Canonical Charmed Kubernetes for AI development on Lenovo ThinkSystem.

Lenovo and Canonical have worked together to build a jointly engineered and validated architecture that details software, hardware, and integration points of all solution components. The architecture provides prescriptive guidance and recommendations for:

- Hardware design
 - Infrastructure nodes
 - Cluster hyperconverged nodes
- Network hardware and design
- Software layout
- System configurations

The intended audience for this document is technical IT architects, system administrators, and managers who are interested in deploying a Canonical Charmed Kubernetes solution on Lenovo ThinkSystem to support AI model development.

48

103. The Lenovo ThinkSystem SR665 performs a method for hot deployment and/or redeployment in a grid computing environment, wherein the grid computing environment includes one or more grid nodes. For example, Kubernetes uses a method of rolling updates to allow for hot deployment and redeployment (rollouts and rollbacks). Using rolling updates, application containers (e.g., pods) are updated and scheduled on worker nodes of the cluster.

⁴⁸ <https://lenovopress.lenovo.com/lp1415.pdf>

Kubernetes progressively rolls out changes to your application or its configuration, while monitoring application health to ensure it doesn't kill all your instances at the same time. If something goes wrong, Kubernetes will rollback the change for you. Take advantage of a growing ecosystem of deployment solutions.

49

When you deploy Kubernetes, you get a cluster.

A Kubernetes cluster consists of a set of worker machines, called nodes, that run containerized applications. Every cluster has at least one worker node.

50

Users expect applications to be available all the time and developers are expected to deploy new versions of them several times a day. In Kubernetes this is done with rolling updates. **Rolling updates** allow Deployments' update to take place with zero downtime by incrementally updating Pods instances with new ones. The new Pods will be scheduled on Nodes with available resources.

51

104. The Lenovo ThinkSystem SR665 performs the step of adding a new version of an application release bundle in a repository server. For example, container images are software bundles stored in a registry. To roll out a new version of software, a container image with the new version is formed.

Once you have a running Kubernetes cluster, you can deploy your containerized applications on top of it. To do so, you create a Kubernetes **Deployment** configuration. The Deployment instructs Kubernetes how to create and update instances of your application. Once you've created a Deployment, the Kubernetes control plane schedules the application instances included in that Deployment to run on individual Nodes in the cluster.

⁴⁹ <https://kubernetes.io/>

⁵⁰ <https://kubernetes.io/docs/concepts/overview/components/>

⁵¹ <https://kubernetes.io/docs/tutorials/kubernetes-basics/update/update-intro/>

When you create a Deployment, you'll need to specify the container image for your application and the number of replicas that you want to run. You can change that information later by updating your Deployment; Modules 5 and 6 of the bootcamp discuss how you can scale and update your Deployments.

52

A container image represents binary data that encapsulates an application and all its software dependencies. Container images are executable software bundles that can run standalone and that make very well defined assumptions about their runtime environment.

You typically create a container image of your application and push it to a registry before referring to it in a Pod.

If you don't specify a registry hostname, Kubernetes assumes that you mean the Docker public registry.

53

105. The Lenovo ThinkSystem SR665 performs the step of determining by a discovery services module which of the one or more grid nodes are running an application associated with the added new version of the application release bundle upon adding the new version of the application release bundle in the repository server. For example, a control plane has a deployment controller that ensures that the new container image is deployed all across the ReplicaSet of the cluster.

Users expect applications to be available all the time and developers are expected to deploy new versions of them several times a day. In Kubernetes this is done with rolling updates. **Rolling updates** allow Deployments' update to take place with zero downtime by incrementally updating Pods instances with new ones. The new Pods will be scheduled on Nodes with available resources.

Initially, the pods run the first version of your application—let's suppose its image is tagged as v1. You then develop a newer version of the app and push it to an image repository as a new image, tagged as v2. You'd next like to replace all the pods with this new version. Because you can't change an existing pod's image after the pod is created, you need to remove the old pods and replace them with new ones running the new image.

54

⁵² <https://kubernetes.io/docs/tutorials/kubernetes-basics/deploy-app/deploy-intro/>

⁵³ <https://kubernetes.io/docs/concepts/containers/images/>

⁵⁴ <https://kubernetes.io/docs/tutorials/kubernetes-basics/update/update-intro/>

A *Deployment* provides declarative updates for Pods and ReplicaSets.

You describe a *desired state* in a Deployment, and the Deployment Controller changes the actual state to the desired state at a controlled rate. You can define Deployments to create new ReplicaSets, or to remove existing Deployments and adopt all their resources with new Deployments.

Each time a new Deployment is observed by the Deployment controller, a ReplicaSet is created to bring up the desired Pods. If the Deployment is updated, the existing ReplicaSet that controls Pods whose labels match `.spec.selector` but whose template does not match `.spec.template` are scaled down. Eventually, the new ReplicaSet is scaled to `.spec.replicas` and all old ReplicaSets is scaled to 0.

55

Once you have a [running Kubernetes cluster](#), you can deploy your containerized applications on top of it. To do so, you create a Kubernetes **Deployment**. The Deployment instructs Kubernetes how to create and update instances of your application. Once you've created a Deployment, the Kubernetes control plane schedules the application instances included in that Deployment to run on individual Nodes in the cluster.

Once the application instances are created, a Kubernetes Deployment controller continuously monitors those instances. If the Node hosting an instance goes down or is deleted, the Deployment controller replaces the instance with an instance on another Node in the cluster. **This provides a self-healing mechanism to address machine failure or maintenance.**

56

106. The Lenovo ThinkSystem SR665 performs the step of notifying a client application manager associated with one or more of the determined grid nodes about adding the new version of the application release bundle along with a type of data transfer protocol to use. The Deployment controller notifies Kubelet processes in worker nodes. Each Kubelet process is therefore notified of the desired configuration, including the new application and associated protocol information.

A *Deployment* provides declarative updates for Pods and ReplicaSets.

⁵⁵ <https://kubernetes.io/docs/concepts/workloads/controllers/deployment/>

⁵⁶ <https://kubernetes.io/docs/tutorials/kubernetes-basics/deploy-app/deploy-intro/>

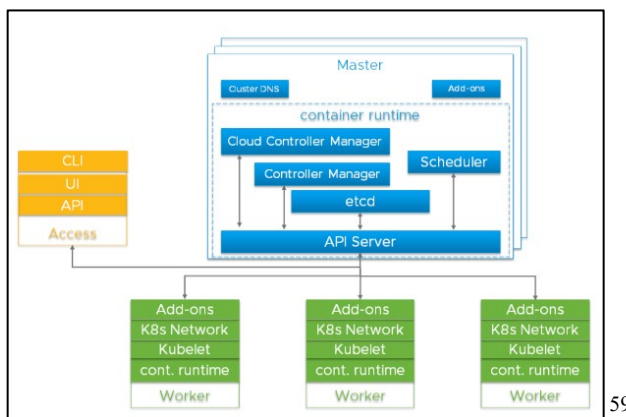
Each time a new Deployment is observed by the Deployment controller, a ReplicaSet is created to bring up the desired Pods. If the Deployment is updated, the existing ReplicaSet that controls Pods whose labels match `.spec.selector` but whose template does not match `.spec.template` are scaled down. Eventually, the new ReplicaSet is scaled to `.spec.replicas` and all old ReplicaSets is scaled to 0.

57

An agent that runs on each node in the cluster. It makes sure that containers are running in a Pod.

The kubelet takes a set of PodSpecs that are provided through various mechanisms and ensures that the containers described in those PodSpecs are running and healthy. The kubelet doesn't manage containers which were not created by Kubernetes.

58



59

⁵⁷ <https://kubernetes.io/docs/concepts/workloads/controllers/deployment/>

⁵⁸ <https://kubernetes.io/docs/concepts/architecture/>

⁵⁹ <https://kubernetes.io/blog/2018/08/03/out-of-the-clouds-onto-the-ground-how-to-make-kubernetes-production-grade-anywhere/>

```

Name:          nginx-deployment
Namespace:     default
CreationTimestamp: Thu, 30 Nov 2017 10:56:25 +0000
Labels:        app=nginx
Annotations:   deployment.kubernetes.io/revision=2
Selector:      app=nginx
Replicas:      3 desired | 3 updated | 3 total | 3 available
StrategyType:  RollingUpdate
MinReadySeconds: 0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=nginx
  Containers:
    nginx:
      Image:   nginx:1.16.1
      Port:    80/TCP
      Environment: <none>
      Mounts:    <none>
      Volumes:   <none>

```

```

Conditions:
  Type           Status  Reason
  ----           -
  Available      True    MinimumReplicasAvailable
  Progressing    True    NewReplicaSetAvailable
OldReplicaSets: <none>
NewReplicaSet:  nginx-deployment-1564180365 (3/3 replicas created)

```

```

Events:
  Type    Reason             Age   From                  Message
  ----    -
  Normal  ScalingReplicaSet  2m    deployment-controller  Scaled up replica set nginx-
  Normal  ScalingReplicaSet  24s    deployment-controller  Scaled up replica set nginx-
  Normal  ScalingReplicaSet  22s    deployment-controller  Scaled down replica set nginx-
  Normal  ScalingReplicaSet  22s    deployment-controller  Scaled up replica set nginx-
  Normal  ScalingReplicaSet  19s    deployment-controller  Scaled down replica set nginx-
  Normal  ScalingReplicaSet  19s    deployment-controller  Scaled up replica set nginx-
  Normal  ScalingReplicaSet  14s    deployment-controller  Scaled down replica set nginx-

```

107. The Lenovo ThinkSystem SR665 performs the step of hot deploying/redeploying the new version of the application release bundle running on one or more application servers in an associated grid node using an appropriate hot deployment plug-in based on the data transfer protocol by a respective one of the client application managers. For example, rolling updates of the new software are made with the rollout command of the Kubelet interface. The image is deployed in the worker nodes according to the desired configuration by the Kubelet process.

⁶⁰ <https://kubernetes.io/docs/concepts/workloads/controllers/deployment/>

Users expect applications to be available all the time and developers are expected to deploy new versions of them several times a day. In Kubernetes this is done with rolling updates. **Rolling updates** allow Deployments' update to take place with zero downtime by incrementally updating Pods instances with new ones. The new Pods will be scheduled on Nodes with available resources.

61

A *Deployment* provides declarative updates for *Pods* and *ReplicaSets*.

You describe a *desired state* in a Deployment, and the Deployment Controller changes the actual state to the desired state at a controlled rate. You can define Deployments to create new ReplicaSets, or to remove existing Deployments and adopt all their resources with new Deployments.

- **Declare the new state of the Pods** by updating the PodTemplateSpec of the Deployment. A new ReplicaSet is created and the Deployment manages moving the Pods from the old ReplicaSet to the new one at a controlled rate. Each new ReplicaSet updates the revision of the Deployment.

- The `template` field contains the following sub-fields:

- The Pods are labeled `app: nginx` using the `.metadata.labels` field.
- The Pod template's specification, or `.template.spec` field, indicates that the Pods run one container, `nginx`, which runs the `nginx` Docker Hub image at version 1.14.2.
- Create one container and name it `nginx` using the `.spec.template.spec.containers[0].name` field.

62

```
Name:          nginx-deployment
Namespace:     default
CreationTimestamp: Thu, 30 Nov 2017 10:56:25 +0000
Labels:        app=nginx
Annotations:   deployment.kubernetes.io/revision=2
Selector:      app=nginx
Replicas:      3 desired | 3 updated | 3 total | 3 available
StrategyType:  RollingUpdate
MinReadySeconds: 0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=nginx
  Containers:
    nginx:
      Image:      nginx:1.16.1
      Port:       80/TCP
      Environment: <none>
      Mounts:      <none>
      Volumes:     <none>
```

⁶¹ <https://kubernetes.io/docs/tutorials/kubernetes-basics/update/update-intro/>

⁶² <https://kubernetes.io/docs/concepts/workloads/controllers/deployment/>

Conditions:		
Type	Status	Reason
----	-----	-----
Available	True	MinimumReplicasAvailable
Progressing	True	NewReplicaSetAvailable
OldReplicaSets:	<none>	
NewReplicaSet:	nginx-deployment-1564180365 (3/3 replicas created)	

Events:				
Type	Reason	Age	From	Message
-----	-----	-----	-----	-----
Normal	ScalingReplicaSet	2m	deployment-controller	Scaled up replica set nginx-
Normal	ScalingReplicaSet	24s	deployment-controller	Scaled up replica set nginx-
Normal	ScalingReplicaSet	22s	deployment-controller	Scaled down replica set nginx-
Normal	ScalingReplicaSet	22s	deployment-controller	Scaled up replica set nginx-
Normal	ScalingReplicaSet	19s	deployment-controller	Scaled down replica set nginx-

63

108. Lenovo directly infringes the '332 Patent through promotional demonstrations, testing, repairs, and instructional guidance of the Lenovo ThinkSystem SR665, performing at least the steps of claim 1.

109. Defendants have and continue to indirectly infringe one or more claims of the '332 Patent by knowingly and intentionally inducing others, including Lenovo customers and end-users, to directly infringe, either literally or under the doctrine of equivalents, by making, using, offering to sell, selling, and/or importing into the United States products that include the infringing technology.

110. Defendants, with knowledge that these products, or the use thereof, infringe the '332 Patent at least as of January 30, 2024,⁶⁴ knowingly and intentionally induced, and continue to knowingly and intentionally induce, direct infringement of the '332 Patent by providing these products to end-users for use in an infringing manner. Additionally, on information and belief, Defendants have adopted a policy of not reviewing the patents of others, including specifically

⁶³ <https://kubernetes.io/docs/concepts/workloads/controllers/deployment/>

⁶⁴ Plaintiffs sent an email to Defendant Lenovo Group Limited on January 30, 2024, notifying them of its infringement of the '332 Patent, among other patents. Therefore, at least Defendant Lenovo Group Limited has been on notice of its infringement of the '332 Patent since at least January 2024.

those related to Defendants' specific industry, thereby remaining willfully blind to the '332 Patent at least as early as the issuance of the '332 Patent.

111. Defendants have induced infringement by others, including end-users, with the intent to cause infringing acts by others or in the alternative with the belief that there was a high probability that others, including end-users, infringe the '332 Patent, but while remaining willfully blind to the infringement. Defendants have and continue to induce infringement by their customers and end-users by supplying them with instructions on how to operate the infringing technology in an infringing manner, while also making publicly available information on the infringing technology via Defendants' website, product literature and packaging, and other publications.

112. Plaintiffs have suffered damages as a result of Defendants' direct and indirect infringement of the '332 Patent in an amount to be proven at trial.

113. Plaintiffs have suffered, and will continue to suffer, irreparable harm as a result of Defendants' infringement of the '332 Patent, for which there is no adequate remedy at law, unless Defendants' infringement is enjoined by this Court.

114. Defendants have committed and continue to commit acts of infringement that Defendants actually knew or should have known constituted an unjustifiably high risk of infringement of at least one valid and enforceable claim of the '332 Patent. Defendants' direct and indirect infringement of the '332 Patent has been and continues to be willful, intentional, deliberate, and/or in conscious disregard of rights under the patent. Plaintiffs are entitled to an award of treble damages, reasonable attorney fees, and costs in bringing this action.

COUNT V
(Infringement of the '509 Patent)

115. Paragraphs 1 through 51 are incorporated by reference as if fully set forth herein.

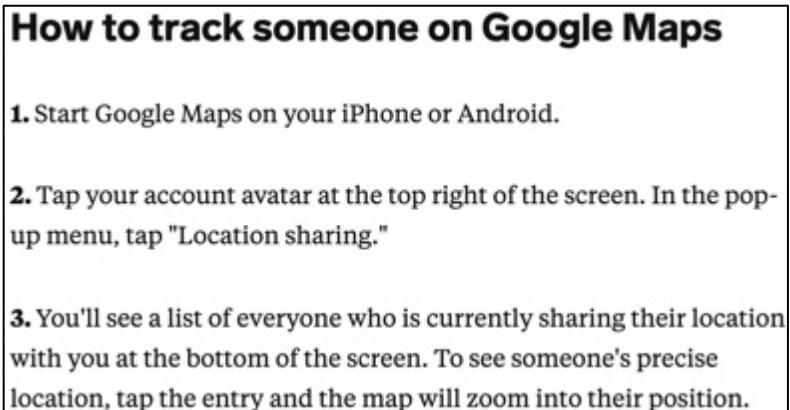
116. Plaintiffs have not licensed or otherwise authorized Defendants to make, use, offer

for sale, sell, or import any products that embody the inventions of the '509 Patent.

117. Defendants have directly infringed the '509 Patent, either literally or under the doctrine of equivalents, without authority and in violation of 35 U.S.C. § 271, by making, using, offering to sell, selling, and/or importing into the United States products that satisfy each and every limitation of one or more claims of the '509 Patent. Such products include smartphones and tablets including, but not limited to, the Moto G Fast, equipped with Google Maps, among other Lenovo smartphones and products.

118. For example, Defendants have directly infringed at least claim 17 of the '509 Patent by making, using, offering to sell, selling, and/or importing into the United States products that include smartphones and tablets including, but not limited to, the Moto G Fast, equipped with Google Maps, among other Lenovo smartphone, tablets, and other computer products.

119. The Moto G Fast performs a method of monitoring an object. For example, the Moto G Fast, through the use of Google Maps, can track the user's location.



65

120. The Moto G Fast performs the step of generating identification data associated with each monitored object located at an identification gateway of a monitored environment, wherein

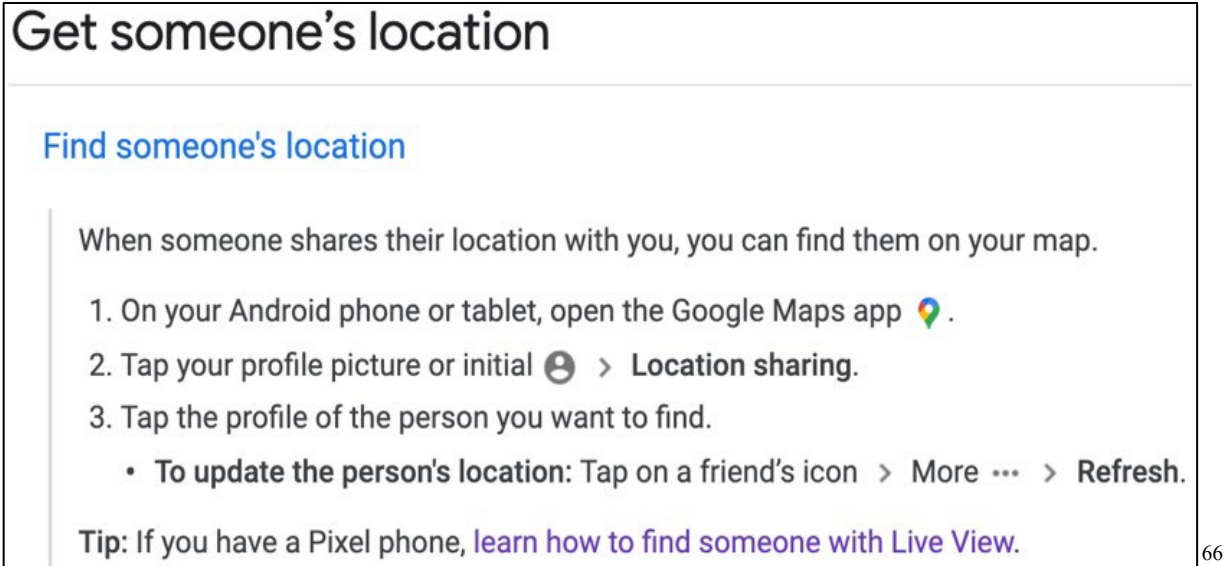
⁶⁵ <https://www.businessinsider.com/how-to-track-someone-on-google-maps#:~:text=You%20can%20track%20someone%20on,location%20tracking%20for%20you%20specifically>

said generating identification data is performed by an identification system. For example, part of the Location Sharing facility of Google Maps as implemented on Lenovo products including, but not limited to, the Moto G Fast, generates the name of each person that is sharing location data, with each person located at an identification gateway, such as a cellular base station or a Wi-Fi access point, of a monitored environment surrounding the person.

121. The Moto G Fast performs the step of generating tracking data associated with each monitored object located within said monitored environment, wherein said generating tracking data is performed by a tracking system. For example, part of the Location Sharing facility of Google Maps as implemented on Lenovo products including, but not limited to, the Moto G Fast, generates the name of each person that is sharing location data.

122. The Moto G Fast performs the step of merging and storing said identification data and said tracking data of each monitored object to form monitoring data for each monitored object. For example, Google Maps merges the name and location of each person that is sharing location data and stores them in the memory of the device running the application.

123. The Moto G Fast performs the step of comparing new content of said monitoring data of a monitored object with prior content of said monitoring data of said monitored object. For example, the Location Sharing facility of Google Maps compares the new location data associated with the name of each person that is sharing location data with the prior location data associated with the same name, mapping the person to a new location if the location has changed.



124. Lenovo has directly infringed the '509 Patent through promotional demonstrations, testing, repairs, and instructional guidance of the Moto G Fast, performing at least the steps of claim 17.

125. Defendants have indirectly infringed one or more claims of the '509 Patent by knowingly and intentionally inducing others, including Lenovo customers and end-users, to directly infringe, either literally or under the doctrine of equivalents, by making, using, offering to sell, selling, and/or importing into the United States products that include the infringing technology.

126. Defendants, with knowledge that these products, or the use thereof, infringed the '509 Patent at least as of February 11, 2022⁶⁷, knowingly and intentionally induced direct

⁶⁶

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&hl=en&oco=1#zippy=%2Cfind-someones-location>

⁶⁷ During the negotiations between Plaintiffs and Defendant Lenovo Group Limited, Plaintiffs sent an email to Defendant Lenovo Group Limited on February 11, 2022, notifying them of its infringement of the '509 Patent. Plaintiffs continued negotiating a license to the '509 Patent, among others, until at least November 2024. Therefore, at least Defendant Lenovo Group Limited has been on notice of its infringement of the '509 Patent since at least February 11, 2022.

infringement of the '509 Patent by providing these products to end-users for use in an infringing manner. Additionally, on information and belief, Defendants have adopted a policy of not reviewing the patents of others, including specifically those related to Defendants' specific industry, thereby remaining willfully blind to the '509 Patent at least as early as the issuance of the '509 Patent.

127. Defendants have induced infringement by others, including end-users, with the intent to cause infringing acts by others or, in the alternative, with the belief that there was a high probability that others, including end-users, infringe the '509 Patent, but while remaining willfully blind to the infringement. Defendants have induced infringement by their customers and end-users by supplying them with instructions on how to operate the infringing technology in an infringing manner, while also making publicly available information on the infringing technology via Defendants' website, product literature and packaging, and other publications.

128. Plaintiffs have suffered damages as a result of Defendants' direct and indirect infringement of the '509 Patent in an amount to be proven at trial.

129. Defendants have committed acts of infringement that Defendants actually knew or should have known constituted an unjustifiably high risk of infringement of at least one valid and enforceable claim of the '509 Patent. Defendants' direct and indirect infringement of the '509 Patent was willful, intentional, deliberate, and/or in conscious disregard of rights under the patent. Plaintiffs are entitled to an award of treble damages, reasonable attorney fees, and costs in bringing this action.

COUNT VI
(Infringement of the '832 Patent)

130. Paragraphs 1 through 51 are incorporated by reference as if fully set forth herein.

131. Plaintiffs have not licensed or otherwise authorized Defendants to make, use, offer for sale, sell, or import any products that embody the inventions of the '832 Patent.

132. Defendants have infringed the '832 Patent, either literally or under the doctrine of equivalents, without authority and in violation of 35 U.S.C. § 271, by making, using, offering to sell, selling, and/or importing into the United States products that satisfy each and every limitation of one or more claims of the '832 Patent. Such products and services include Lenovo servers, storage systems, data centers, e-commerce, and other systems that Lenovo employs that use Apache Kafka Streams using RocksDB Universal Compaction and also systems using Apache Cassandra, such as the Lenovo ThinkSystem SR665, among other Lenovo products and services.

133. For example, Defendants have directly infringed at least claim 25 of the '832 Patent by making, using, offering to sell, selling, and/or importing into the United States products that include Lenovo servers, storage systems, data centers, e-commerce, and other systems that use Apache Kafka Streams using RocksDB Universal Compaction and also systems using Apache Cassandra, such as the Lenovo ThinkSystem SR665, among other Lenovo products.

134. Upon information and belief, Lenovo products use Kafka Streams.

135. Upon information and belief, Lenovo employees use Kafka Streams, as depicted below.

Full-stack Engineer

Lenovo - Morrisville, NC
August 2021 to October 2022

Responsibilities:

- Implemented user interface coding and styling using ReactJS, CSS, Node.js, and Bootstrap template.
- Created responsive web design and developed a single ISOMORPHIC responsive website that could be served to desktop, tablets, and mobile users using React.
- Expertise in using React.js/Redux to build User Interfaces, strong knowledge of state store, middleware, action creator, reducer, and container.
- Created responsive design and developed a single responsive website that could be served to desktop, Tablet, and mobile users using ReactJS.
- REACT JS Virtual DOM used for client-side view rendering services, React-Redux for state management, and React-Router for programmatic navigation.
- Built servers and web services in Node.js and Express.js interacting with MongoDB using Mongoose schema.
- A user provision script developed using NodeJS which includes fetching files from the SFTP server and creating users in Idap and database using REST API calls.
- Used Node.js to structure JavaScript code to build Restful web services. Implemented generating the verification hash code using a crypto package on Node.js.
- Creating script for data modeling and data import and export. Extensive experience in deploying, managing, and developing MongoDB clusters.
- Designed with MongoDB to store non-relational data in collection and retrieve them whenever required.
- Implemented scripts for Mongo DB import, export, dump, and restore. Worked on MongoDB database concepts such as locking, transactions, indexes, Sharding, replication, schema design
- Implemented scripts for Mongo DB import, export, dump, and restore. Worked on MongoDB database concepts such as locking, transactions, indexes, Sharding, replication, and schema design.
- Used Serverless services, created and configured HTTP Triggers in the Azure Functions with application insights for monitoring and performing load testing on the applications using the Visual Studio Team Services (VSTS) also called Azure DevOps Services.
- Created CI/CD Pipelines in Azure DevOps environments by providing their dependencies and tasks. Also, have experience in implementing and managing continuous delivery systems and methodologies on AWS and created END-END Automation with CI Procedures using Jenkins & automated Maven builds by integrating them with Continuous Integration tools Jenkins.
- Implemented multiple Kafka data pipelines handling billions of events daily to enable real-time stream processing and analytics.
- Backlog tracking & Status Reporting to Key Stakeholders using Azure DevOps.

68

Kafka Streams

136. Kafka Streams used by Lenovo uses RocksDB Universal Compaction to perform a method for monitoring performance of a storage device.

Optimized for Fast Storage

RocksDB is optimized for fast, low latency storage such as flash drives and high-speed disk drives. RocksDB exploits the full potential of high read/write rates offered by flash or RAM.

⁶⁸ <https://resumes.indeed.com/resume/ed2b4d983a41370c>

RocksDB provides several APIs to read KV pairs from a database, including Get and MultiGet for point lookups and Iterator for sequential scanning. These APIs may result in RocksDB reading blocks from SST files on disk storage. The types of blocks and the frequency with which they are read from storage is workload dependent. Some workloads may have a small working set and thus may be able to cache most of the data required, while others may have large working sets and have to read from disk more often. In the latter case, the latency would be much higher and throughput would be lower than the former. They would also be dependent on the characteristics of the underlying storage media, making it difficult to migrate from one medium to another, for example, local flash to disaggregated flash.

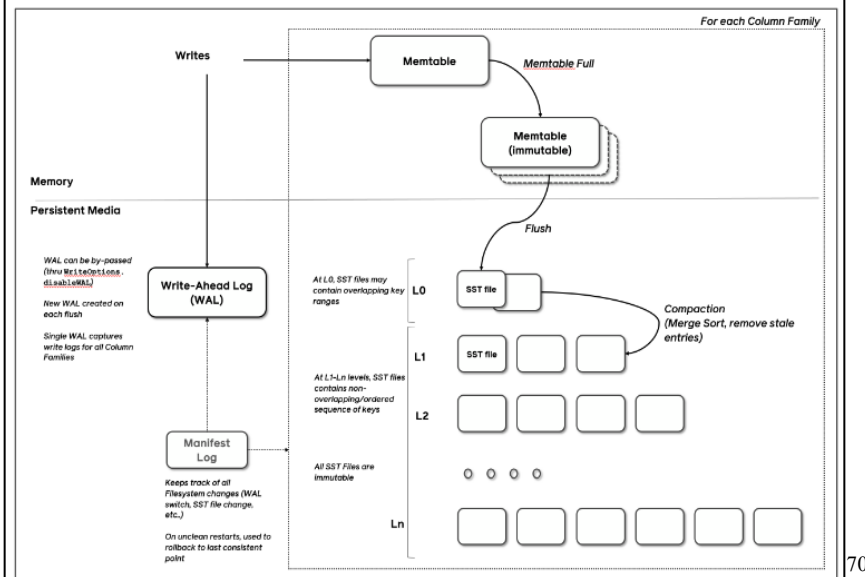
69

3. High Level Architecture

RocksDB is a storage engine library of key-value store interface where keys and values are arbitrary byte streams. RocksDB organizes all data in sorted order and the common operations are `Get(key)`, `NewIterator()`, `Put(key, val)`, `Delete(key)`, and `SingleDelete(key)`.

The three basic constructs of RocksDB are **memtable**, **sstfile** and **logfile**. The **memtable** is an in-memory data structure - new writes are inserted into the **memtable** and are optionally written to the **logfile (aka. Write Ahead Log(WAL))**. The logfile is a sequentially-written file on storage. When the memtable fills up, it is flushed to a **sstfile** on storage and the corresponding logfile can be safely deleted. The data in an **sstfile** is sorted to facilitate easy lookup of keys.

The default format of **sstfile** is described in more details [here](#).



70

⁶⁹ <https://rocksdb.org/blog/>

⁷⁰ <https://github.com/facebook/rocksdb/wiki/RocksDB-Overview>

Compaction Styles

Both Level Style Compaction and Universal Style Compaction store data in a fixed number of logical *levels* in the database. More recent data is stored in Level-0 (L0) and older data in higher-numbered levels, up to Lmax. Files in L0 may have overlapping keys, but files in other levels generally form a single sorted run per level.

Level Style Compaction (default) typically optimizes disk footprint vs. logical database size (space amplification) by minimizing the files involved in each compaction step: merging one file in L_n with all its overlapping files in L_{n+1} and replacing them with new files in L_{n+1} .

Universal Style Compaction typically optimizes total bytes written to disk vs. logical database size (write amplification) by merging potentially many files and levels at once, requiring more temporary space. Universal typically results in lower write-amplification but higher space- and read-amplification than Level Style Compaction.

71

137. On information and belief, by using Kafka Streams using RocksDB Universal Compaction, Lenovo performs the step of intercepting communications between a computer system and said storage device.

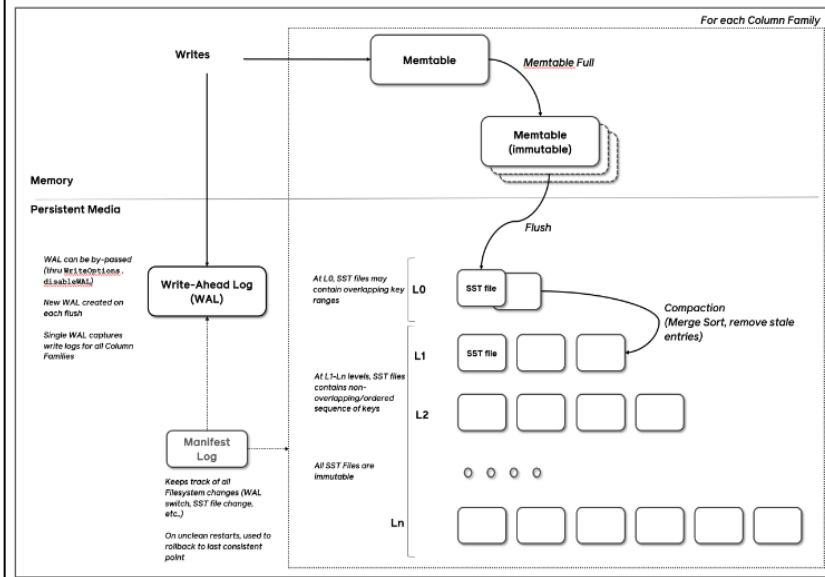
⁷¹ <https://github.com/facebook/rocksdb/wiki/RocksDB-Overview>

3. High Level Architecture

RocksDB is a storage engine library of key-value store interface where keys and values are arbitrary byte streams. RocksDB organizes all data in sorted order and the common operations are `Get(key)`, `NewIterator()`, `Put(key, val)`, `Delete(key)`, and `SingleDelete(key)`.

The three basic constructs of RocksDB are **memtable**, **sstfile** and **logfile**. The **memtable** is an in-memory data structure - new writes are inserted into the **memtable** and are optionally written to the **logfile** (aka. **Write Ahead Log(WAL)**). The logfile is a sequentially-written file on storage. When the memtable fills up, it is flushed to a **sstfile** on storage and the corresponding logfile can be safely deleted. The data in an sstfile is sorted to facilitate easy lookup of keys.

The default format of sstfile is described in more details [here](#).



72

138. On information and belief, by using Kafka Streams using RocksDB Universal Compaction, Lenovo performs the step of analyzing said intercepted communications.

3. High Level Architecture

RocksDB is a storage engine library of key-value store interface where keys and values are arbitrary byte streams. RocksDB organizes all data in sorted order and the common operations are `Get(key)`, `NewIterator()`, `Put(key, val)`, `Delete(key)`, and `SingleDelete(key)`.

The three basic constructs of RocksDB are **memtable**, **sstfile** and **logfile**. The **memtable** is an in-memory data structure - new writes are inserted into the **memtable** and are optionally written to the **logfile** (aka. **Write Ahead Log(WAL)**). The logfile is a sequentially-written file on storage. When the memtable fills up, it is flushed to a **sstfile** on storage and the corresponding logfile can be safely deleted. The data in an sstfile is sorted to facilitate easy lookup of keys.

73

⁷² <https://github.com/facebook/rocksdb/wiki/RocksDB-Overview>

⁷³ <https://github.com/facebook/rocksdb/wiki/Universal-Compaction>

Precondition: $n \geq \text{options.level0_file_num_compaction_trigger}$

Unless number of sorted runs reaches this threshold, no compaction will be triggered at all.

If the pre-condition is satisfied, there are four conditions. Each of them can trigger a compaction. They are evaluated in order and the selection stops once a compaction has been scheduled. Before going into detail, the overview for picking a compaction is:

1. Try to schedule a compaction by age of data
2. Else if the space amplification constraint has been violated, try to schedule a major compaction
3. Else try to schedule a compaction (minor or major) while respecting [size_ratio](#)
4. Else try to schedule a minor compaction without respecting [size_ratio](#)

74

1. Compaction triggered by age of data

For universal style compaction, the aging-based triggering criterion has the highest priority since it is a hard requirement. When trying to pick a compaction, this condition is checked first.

If the condition meets (there are files older than `options.periodic_compaction_seconds`), then RocksDB proceeds to pick sorted runs for compaction. RocksDB picks sorted runs from oldest to youngest until encountering a sorted run that is being compacted by another compaction. These files will be compacted to bottommost level unless bottommost level is reserved for ingestion behind. In this case, files will be compacted to second bottommost level.

2. Compaction Triggered by Space Amplification

If the estimated *size amplification ratio* is larger than `options.compaction_options_universal.max_size_amplification_percent / 100`, all files will be compacted to one sorted run.

3. Compaction Triggered by number of sorted runs while respecting *size_ratio*

If a compaction is not triggered by the space amplification constraint as described above then a compaction can be triggered by the number of sorted runs. This step respects *size_ratio trigger* when selecting the sorted runs to compact.

We calculated a value of *size_ratio trigger* as

```
size_ratio_trigger = (100 + options.compaction_options_universal.size_ratio) / 100
```

4. Compaction Triggered by number of sorted runs without respecting *size_ratio*

If none of the previous conditions cause a compaction to be scheduled, then this condition might schedule one. This step does not respect *size_ratio trigger* but the logic is otherwise similar to what was described above for condition 3. If we need to compact more than `options.compaction_options_universal.max_merge_width` number of sorted runs, we cap it to `options.compaction_options_universal.max_merge_width`.

75

139. On information and belief, by using Kafka Streams using RocksDB Universal Compaction, Lenovo performs the step of analyzing said intercepted communications reallocating at least some of said data on said storage device to enhance the performance of said storage device

⁷⁴ <https://github.com/facebook/rocksdb/wiki/Terminology>

<https://github.com/facebook/rocksdb/wiki/Universal-Compaction>

⁷⁵ <https://github.com/facebook/rocksdb/wiki/Universal-Compaction>

based on said analyzed communications.

In the presence of ongoing writes, compactions are needed for space efficiency, read (query) efficiency, and timely data deletion. Compaction removes key-value bindings that have been deleted or overwritten, and re-organizes data for query efficiency. Compactions may occur in multiple threads if configured.

The entire database is stored in a set of *sstfiles*. When a *memtable* is full, its content is written out to a file in Level-0 (L0) of the LSM tree. RocksDB removes duplicate and overwritten keys in the memtable when it is flushed to a file in L0. In *compaction*, some files are periodically read in and merged to form larger files, often going into the next LSM level (such as L1, up to Lmax).

The overall write throughput of an LSM database directly depends on the speed at which compactions can occur, especially when the data is stored in fast storage like SSD or RAM. RocksDB may be configured to issue concurrent compaction requests from multiple threads. It is observed that sustained write rates may increase by as much as a factor of 10 with multi-threaded compaction when the database is on SSDs, as compared to single-threaded compactions.

76

Apache Cassandra

140. Upon information and belief, Lenovo employees and products use Apache Cassandra, as depicted below.

⁷⁶ <https://github.com/facebook/rocksdb/wiki/RocksDB-Overview>

Sr. Hadoop Administrator

Lenovo, LosAngeles, CA

July 2015 to Present

Duration: 13 months

Responsibilities and Achievements:

- Install and Manage HDP Hortonworks DL and DW components.
- Worked on Hadoop Hortonworks (HDP 2.6.0.2.2) distribution which managed services viz. HDFS, MapReduce2, Tez, Hive, Pig, Hbase, Sqoop, Flume, Spark, Ambari Metrics, Zookeeper, Falcon and oozie etc.) for 4 cluster ranges from LAB, DEV, QA to PROD contains nearly 350+ nodes with 7PB data.
- Monitor Hadoop cluster connectivity and security on Ambari monitoring system.
- Led the installation, configuration and deployment of product soft wares on new edge nodes that connect and contact Hadoop cluster for data acquisition.
- Rendered L3/L4 support services for BI users, Developers and Tableau team through Jira ticketing system.
- One of the key engineers in Aetna's HDP web engineering team, Integrated Systems engineering ISE.
- Managed and reviewed Log files as a part of administration for troubleshooting purposes. Communicate and escalate issues appropriately for the tickets raised by users in JIRA ticketing system
- Worked closely with System Administrators, BI analysts, developers, and key business leaders to establish SLAs and acceptable performance metrics for the Hadoop as a service offering.
- Performance Tuning and ETL, Agile Software Deployment, Team Building & Leadership, Engineering Management.
- Hortonworks Ambari, Apache Hadoop on Redhat, and Centos as data storage, retrieval, and processing systems.
- Setting up Kerberos principals in KDC server and testing HDFS, Hive, Pig and MapReduce access for the new users and creating key tabs for service ID's using keytab scripts.
- Performed a Major upgrade in production environment from HDP 2.3 to HDP 2.6. As an admin followed standard Back up policies to make sure the high availability of cluster.
- Monitored multiple Hadoop clusters environments using Ganglia and Nagios. Monitored workload, job performance and capacity planning using Ambari.
- Installed OLAP software Atscale on its designated edge node server.
- Implemented dual data center set up for all Cassandra cluster. Performed much complex system analysis in order to improve ETL performance, identified high critical batch jobs to prioritize.
- Conducted cluster sizing, tuning, and performance benchmarking on a multi-tenant OpenStack platform to achieve desired performance metrics.
- Good knowledge on providing solution to the users who encountered java exception and error problems while running the data models in SAS script and Rscript. Good understanding on forest data models.
- Worked on data ingestion on systems to pull data scooping from traditional RDBMS platforms such as Oracle, MySQL and Teradata to Hadoop cluster using automated ingestion scripts and also store data in NoSQL databases such as HBase, Cassandra.

77

⁷⁷ <https://resumes.indeed.com/resume/c4ccc2153b7db2e2>

Sr. PERFORMANCE ENGINEER/LEAD

LENOVO-TCS - Chicago, IL
July 2016 to Present

Team size: 11)

Project Description:Lenovo has operations in more than 60 countries and sells its products in around 160 countries. Lenovo's principal facilities are in Beijing and Morrisville. All these countries websites are built on e-comm ongoing, Performance & optimization environment. My role is to perform load, regression, sanity, firebug test and analyze the bottlenecks to fix them. Also, Involve in POC workload model.

Responsibilities:

- Conducted work group sessions with application managers, developers, Business analyst for gathering requirements, analyzed requirements for creating work load model to get ready for on coming Holiday readiness.
- Created build analysis for mobile applications using Fiddler4 (Identify unnecessary JSON/API calls and other functionality issues) and convert to LoadRunner Test Scripts.
- Used Appium for mobile automation testing using Android and IOS devices.
- Performance tuning is a major part of days activity, Troubleshooting performance issues on production databases and work with developers to tune their queries in Test/Dev systems.
- Responsible for the creation of Confluence page for test plan/strategy, test schedule, testing status reporting, test case creation, monitoring of test case execution and script execution where needed.
- Write/maintain test case scripts and execute; document detailed results and summary report.
- Developed Test plans to ensure accomplishment of load-testing objectives.
- Developed and Executed Jmeter/Blazemeter Scripts.
- Leading and coordinating with three offshore resources to deliver the SAP Performance solution
- Coordination of Execution of the scripts and reporting the bugs through defect tracking tool.
- Coordination and Creation of Design of SAP Web and SAP GUI scripts
- Established 150 AWS EC2 load generators utilizing the ALM Performance Center controllers; maintained EC2 images and start/stop cloud machine scripts.
- Perform Mobile test using different mobile testing devices and different platform like IOS, Android and mobile web.
- Worked on tickets/issues opened in Issue Management Trackers, CI/CD for deploy builds to various test environments.
- Used IBM rational performance test, team center for web application or server to a large volume of transactions for the purpose of measuring the server response times.
- Involved in test environment build and designed Load (capacity) model based on current volume and projected percentage increase in volume.
- create detailed System Test plans and to participate in reviews of System Test plans, System Test designs and System Test automation.
- Extended the Support to the users of Retail POS to Dynamics CRM in their day to day activities
- Performed backend development using open source toolset (PHP, MySQL, Apache, Linux and others (i.e LAMP)
- Integrated Performance Testing with various applications as well as within a Cloud environment.
- Build HP-StormRunner Load Tests (Scenarios) for cloud performance testing.
- Designed XML schema definitions (XSDs) to support the mapping of standard TIBCO Active Enterprise message payloads onto SOAP API calls.
- Performed Baseline (sanity) test, stress test and high volume of users (capacity planning) using JMeter/Blazemeter and monitored the performance of the load test on the system and measured database response time, Http request, Login and proxy server.
- Extensively Worked in Web (HTTP/HTML), True client, Webservices, mobile protocols. Reviewed scripts that were developed by teammates and verified scripts in the standalone mode.
- Extensively used JVisual VM to Monitor the JVM for CPU, Heap, GC, Thread behavior and I/O Stat using UNIX commands like top, Vmstat, Ibmom TCP/IP and Net stat while system under test.

- POC on integrating performance center with Jenkins using backend REST APIs, front JSON to support Continues Testing and Continuous Integration.
 - Experience with JProfiler to identify memory leak in .NET application
 - Uploaded Scripts, Created Timeslots, Created Scenarios, maintained scripts and Run the Load Tests in performance Center v12.5. Analyzed Test Results Response time, Transaction per Seconds and Throughput per graphs.
 - Installed, Configured, Managed Monitoring Tools such as Splunk, Dynatrace for Resource Monitoring/Network Monitor/Log Trace Monitoring.
 - Creating and debugging performance test script using HP LoadRunner 11 VuGen Component with Web (HTTP/HTML), Web Services, Citrix and Ajax TrueClient.
 - Analyzing performance reports to help with root cause analysis (RCA)
 - Extensively worked on GIT for the project maintenance
 - Used JIRA/Atlassian, HP ALM, to Create, Watch and resolve P1, P2, P3 issues assigned to me or to the team as a defect management tool.
 - Developed Test scripts through LoadRunner VUGen and Created Different Scenarios for each Application, executing them in ALM.
 - Used IP spoofing to generate and associate different IP addresses to Virtual Users to emulate real-time scenarios for load balancing issues.
 - Identify optimization opportunities using a range of tools (including YSlow, Pagespeed, LoadRunner, JMeter, Dynatrace, visual inspection of code etc)
 - Analyzed Load Runner on-line graphs and reports to identify network/client delays, CPU/memory usage, I/O delays, database locking and other issues at server level.
 - Responsible for Performance Tuning .NET Application.
 - Experience in installation, configuration and troubleshooting of Web Sphere, Apache Tomcat, JBoss, cloudfoundry, and BEA WebLogic, LDAP, and mail servers.
 - Configured Performance Monitors to monitor and analyze the performance of the server by generating various reports from CPU utilization, Memory Usage to load average etc.
 - Analyzed LoadRunner Metrics and other performance monitoring tools results during and after performance testing on Application server database and generated various Graphs and Reports.
 - Involved in end to end testing like System, Functional, Regression, Smoke, Data Migration, User Acceptance (UAT) Testing.
 - Utilized Database, Network, Application server and WebLogic Monitors during the execution to identify bottlenecks, bandwidth problems, infrastructure problems, and scalability and reliability benchmarks.
 - Created Dashboards in Dynatrace Diagnostics, Daily production analysis reports, Adding Dynatrace headers in each Test scripts for analysis to find the performance bottlenecks.
 - Tuned catalog servers (ehcache) with JDK tools; tested Mobile web APIs, eDialog mail WebSphere MQ messages in JMeter/Blazemeter.
 - Installed, configured, tested OS architecture, Kernel, tuning, Networking, Various Hardware Configs impacting performance of the applications and can provide suggestions based on statistics
 - Produced and discuss closure results with BA teams for their SLA acceptance.
 - Conducted performance testing cycle for new integration framework platform via Dynatrace diagnostic tool to detect performance issues; launched new retail webstores.
 - Executing complex SQL queries (Oracle and MS SQL) and NoSQL (Cassandra, MongoDB) . Monitored Application Server through Analysis. Analyzed various graphs by LoadRunner Analysis and communicated bottleneck issues to the System administrator
 - Responsible for complete functional and regression testing, application used to manage and maintain the leasing data in the company.
- Environment: HP-StormRunner, Load runner 12/12.02, Jmeter 2.8, 2.9, JAVA, J2EE, Linux, Vmware, Cloud architecture, Docker, Performance center 12.5/ALM, Dyna Trace 6.1, Web logic 11g, Windows XP, VUgen, Jmeter, Web methods Integration Servers, Windows 2008, Windows Vista, Web applications, Portal applications, XML files, Jconsole, SOAP.

78

⁷⁸ <https://resumes.indeed.com/resume/4fa925d650742e8a>

Advisory IT Engineer

Lenovo

July 2019 to Present

Virtual office since March 2020)

Serve as technical lead in the DevOps Center of Excellence, Data Platform and Engineering Enablement Group within the Lenovo IT organization. Build tools and solutions to improve engineering efficiency and achieve cost savings across company's IT development teams.

- Product owner for the Performance Test as a Service scrum team, designed and architected product features, led development of the platform. The roll-out of the JMeter and AWS DevOps automation-based product has saved \$120K software license cost spent annually and eliminated ~ \$500K worth of manual work on setting up test environments every year.
- Implemented Selenium based test automation framework into the standard DevOps pipeline in order to automate quality validation during each software build cycle.

Staff IT Engineer

Motorola Mobility - Chicago, IL

January 2015 to June 2019

Served as technical lead in Motorola Mobility eCommerce IT team, developed and supported cloud-based eCommerce microservices and database solutions. Drove business and process improvements using data analysis and BI reporting.

- Drove results with data analysis leveraging BigQuery tool. Designed and developed detailed analytics to identify consumer order volume patterns, errors, and exceptions in real time. The solution reduced order processing cycle time by 50%.
- Designed and implemented distributed database foundation for Motorola eCommerce, adopted NoSQL Cassandra (DataStax Enterprise) database and enabled applications living on elastic and high available cloud infrastructure (AWS) with zero downtime and zero spend on disaster recovery infrastructure.
- Developed cost control process and implemented timely reporting to manage cloud infrastructure cost. Drove teams to reduce AWS spending by 75% since 2016.
- Established system administration, maintenance, and trouble-shooting procedures for Cassandra. Trained the IT Operations team to utilize these procedures to make sure the database cluster stays healthy and high performance.
- Developed Java application with microservices based architecture, which breaks down business functions into independent deployable services so that they can be scaled up and down individually based on need.
- Implemented Omni channel merchandising by integrating Motorola.com with eBay and Amazon online channels. Increased B2C eCommerce order velocity through-put by 10%.

79

⁷⁹ <https://resumes.indeed.com/resume/bed52d8b2a3739c7>

Senior Cloud Performance Engineer

Lenovo - Raleigh-Durham, NC
March 2017 to Present

Tools: JMeter, StressStimulus, Soasta, Dynatrace, AppDynamics, Grafana, Selenium, Cloud Watch, IBM Memory Analyzer,

Technology: Hybris, Microservices, Informatica MDM, MS D365, NodeJS, Java, AWS Cloud, Hana, MySQL, Kafka

Key Responsibilities

- End to end performance engineering, holiday preparation and year-around releases certifications
- Work with stakeholders for KPI's, NFR's and build workload model and performance strategy.
- Review architecture, design for performance and understand application Architecture.
- Performance issue root cause analysis, unit performance tests & code optimization, monitoring & capacity management, implement proof of concepts on the fly improving Performance standards.
- Develop programs using Soasta/Load runner tools on Web, RDP, Ajax True client, Oracle Apps & java protocols, enhance programs for modularity, re-usability.
- Execute load, stress, and endurance tests, identify performance bottlenecks, and certify releases
- Holistic performance analysis and providing recommendations for existing production system/issues
- Install, maintain, and implement all performance testing tools in right environments; including all upgrades and patches, improve performance engineering process
- Work with Dev & Devops regarding on defining hardware requirements, performance CI & CD for continuous integration, resolve integration and environment issues,
- Production analysis of performance, availability and capacity using tools like AppDynamics, Dynatrace, Splunk and create dashboards and alerts for monitoring and alerting possible issues.
- Identify, implement, and lead best practices, and methodologies for an automated/performance testing environment in driving right test results. and contribute in all initiatives to improve software and hardware performance.
- Code review and profile software system based on Micro services, Enterprise Java, JSP, Java Script, jQuery, Spring, Hibernate
- Database monitoring, SQL analysis for Oracle, MySql & Cassandra databases
- WebLogic tuning, thread dump, heap dump analysis, reviewing Oracle AWR reports
- Attend daily scrums, Partner with Product Owners to discuss and resolve all performance issues.

80

Sr. Wintel/Windows/Vmware Admin/Engg

Lenovo - Mooresville, NC
November 2016 to Present

Duties:

- ❖ Installed, Configured, maintained and troubleshoot the Windows Server's 2003, 2008 R2 and 2012 R2 windows servers.
- ❖ Worked with CommVault Backup & Netapp support backup system and team to recover and rebuilt the failover servers for business continuity. Worked on HP ProLiant, Cisco UCS, Dell.
- ❖ Accessed these servers through HP ILO to install new OS or restore from backups on CommVault and Symantec.
- ❖ Worked on rebuilding physical servers from Windows 2003 and 2008 to Windows 2012 R2 datacenter version.
- ❖ Maintain Active Directory (DHCP, WINS & DNS) within the networking environment (LAN) of Windows 2008.
- ❖ Perform Microsoft Server 2012 maintenance and upgrades.

- ❖ Supported and worked on Vrealize, Vrealize operations, Vrealize log insight, Vrealize Automatin.
- ❖ vRealize Operations Manager 6.0 implementation across datacenters.
- ❖ Handled all aspects of rolling upgrade tests and identified critical bugs between the interface of Cassandra Database management (configuration repository) and vRealize Log Insight.
- ❖ Configured vCenter and ESXi for vRealize Automation and NSX integrated environment.
- ❖ Manage over 2200 VM's in VMware vSphere & OVM infrastructure.
- ❖ Coordinate with customers, application, database, network and hardware teams to support and troubleshoot problems and issues. Migrated servers such as File servers, Applications servers, Web servers and Database servers to new OS.
- ❖ Migrated from Webservers and websites from IIS 6.0, IIS 7.0, IIS 7.5 to IIS 8.0.

81

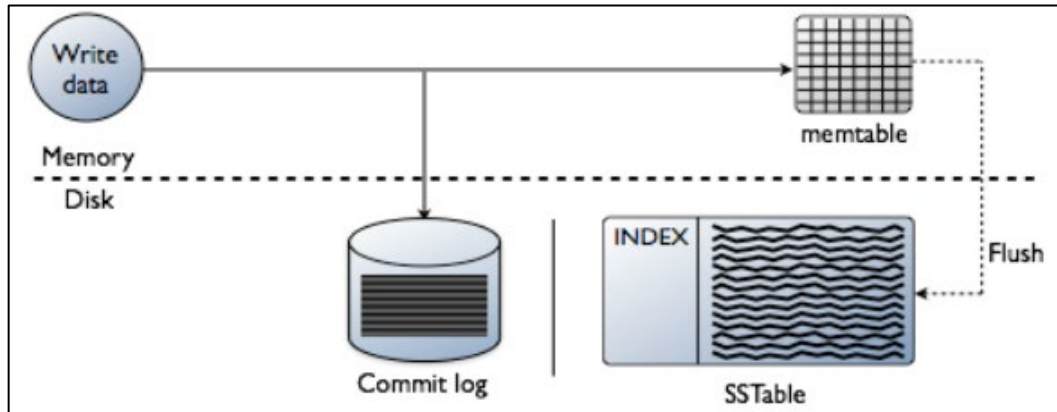
⁸⁰ <https://resumes.indeed.com/resume/97086d2c9f082a59>

⁸¹ <https://resumes.indeed.com/resume/c8666a9b19cc8924>

141. Apache Cassandra used by Lenovo performs a method for monitoring performance of a storage device. For example, Apache Cassandra monitors performance metrics and uses a process called compaction.

- `commitlog_total_space` : Total space to use for commit logs on disk.

82



83

Cassandra Compaction is a process of reconciling various copies of data spread across distinct SSTables. Cassandra performs compaction of SSTables as a background activity. Cassandra has to maintain fewer SSTables and fewer copies of each data row due to compactions improving its read performance. Compaction is a crucial area of Cassandra performance and maintenance.

⁸² https://cassandra.apache.org/doc/stable/cassandra/architecture/storage_engine.html

⁸³ <https://docs.datastax.com/en/cassandra-oss/3.0/cassandra/dml/dmlHowDataWritten.html>

Cassandra's Write Path

To understand the importance of compactions in Cassandra, you must first understand how Cassandra writes data to a disk. The Cassandra write path in a nutshell:

1. Cassandra stores recent writes in memory (in a structure called the Memtable).
2. When enough writes have been made, Cassandra flushes the Memtable to disk. Data on disk is stored in relatively simple data structures called Sorted String Tables (SSTable). At the most simplified level, an SSTable could be described as a sorted array of strings.
3. Before writing a new SSTable, Cassandra merges and pre-sorts the data in the Memtable according to Primary Key. In Cassandra, a Primary Key consists of a Partition Key (the unique key that determines which node the data is stored on) and any Clustering Keys that have been defined.
4. The SSTable is written to disk as a single contiguous write operation. SSTables are immutable. Once they are written to disk they are not modified. Any updates to data or deletion of data within an SSTable is written to a new SSTable. If data is updated regularly, Cassandra may need to read from multiple SSTables to retrieve a single row.
5. Compaction operations occur periodically to re-write and combine SSTables. This is required because SSTables are immutable (no modifications once written to disk). Compactions prune deleted data and merge disparate row data into new SSTables in order to reclaim disk space and keep read operations optimized.

If you are unfamiliar with Cassandra's write path, please read [The write path to compaction](#) from Cassandra Wiki.

84

ReadLatency	Latency	Local read latency for this table.
-------------	---------	------------------------------------

85

⁸⁴ <https://www.instaclustr.com/blog/apache-cassandra-compaction/>

⁸⁵ <https://cassandra.apache.org/doc/stable/cassandra/operating/metrics.html>

Compaction

Strategies

Picking the right compaction strategy for your workload will ensure the best performance for both querying and for compaction itself.

Size Tiered Compaction Strategy (STCS)

The default compaction strategy. Useful as a fallback when other strategies don't fit the workload. Most useful for non pure time series workloads with spinning disks, or when the I/O from `LCS` is too high.

Leveled Compaction Strategy (LCS)

Leveled Compaction Strategy (LCS) is optimized for read heavy workloads, or workloads with lots of updates and deletes. It is not a good choice for immutable time series data.

Time Window Compaction Strategy (TWCS)

Time Window Compaction Strategy is designed for TTL'ed, mostly immutable time series data.

86

Types of compaction

The concept of compaction is used for different kinds of operations in Cassandra, the common thing about these operations is that it takes one or more SSTables and output new SSTables. The types of compactions are:

Minor compaction

triggered automatically in Cassandra.

Major compaction

a user executes a compaction over all SSTables on the node.

User defined compaction

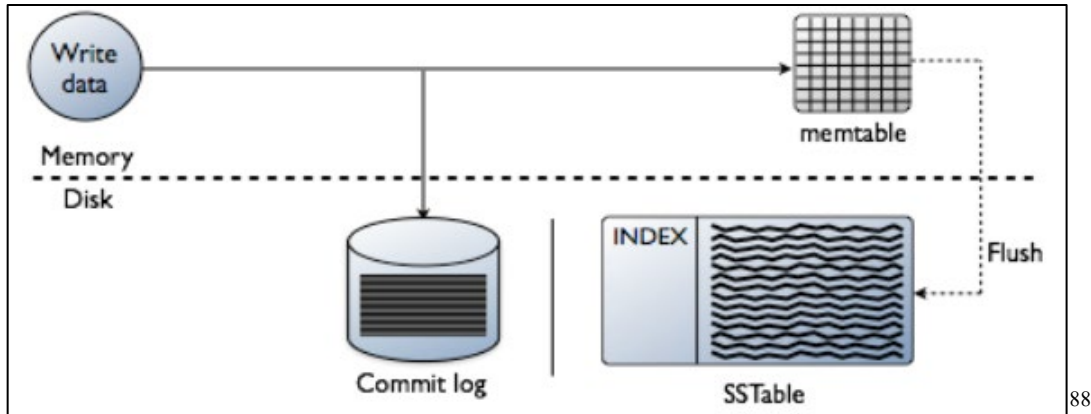
a user triggers a compaction on a given set of SSTables.

87

142. On information and belief, by using Apache Cassandra, Lenovo performs the step of intercepting communications between a computer system and said storage device. For example, when writing data using Apache Cassandra, the step of intercepting communications between a computer (e.g., a Cassandra node) and a storage device (e.g., a disk associated with the Cassandra node) entails populating data into a memtable.

⁸⁶ <https://cassandra.apache.org/doc/stable/cassandra/operating/compaction/>

⁸⁷ <https://cassandra.apache.org/doc/latest/cassandra/operating/compaction/>



143. On information and belief, by using Apache Cassandra, Lenovo performs the step of analyzing said intercepted communications.

Common options

There is a number of common options for all the compaction strategies;

enabled (default: true)

Whether minor compactions should run. Note that you can have 'enabled': true as a compaction option and then do 'nodetool enableautocompaction' to start running compactions.

tombstone_threshold (default: 0.2)

How much of the sstable should be tombstones for us to consider doing a single sstable compaction of that sstable.

tombstone_compaction_interval (default: 86400s (1 day))

Since it might not be possible to drop any tombstones when doing a single sstable compaction we need to make sure that one sstable is not constantly getting recomacted - this option states how often we should try for a given sstable.

log_all (default: false)

New detailed compaction logging, see [below <detailed-compaction-logging>](#).

unchecked_tombstone_compaction (default: false)

The single sstable compaction has quite strict checks for whether it should be started, this option disables those checks and for some usecases this might be needed. Note that this does not change anything for the actual compaction, tombstones are only dropped if it is safe to do so - it might just rewrite an sstable without being able to drop any tombstones.

only_purge_repaired_tombstone (default: false)

Option to enable the extra safety of making sure that tombstones are only dropped if the data has been repaired.

min_threshold (default: 4)

Lower limit of number of SSTables before a compaction is triggered. Not used for `LeveledCompactionStrategy`.

max_threshold (default: 32)

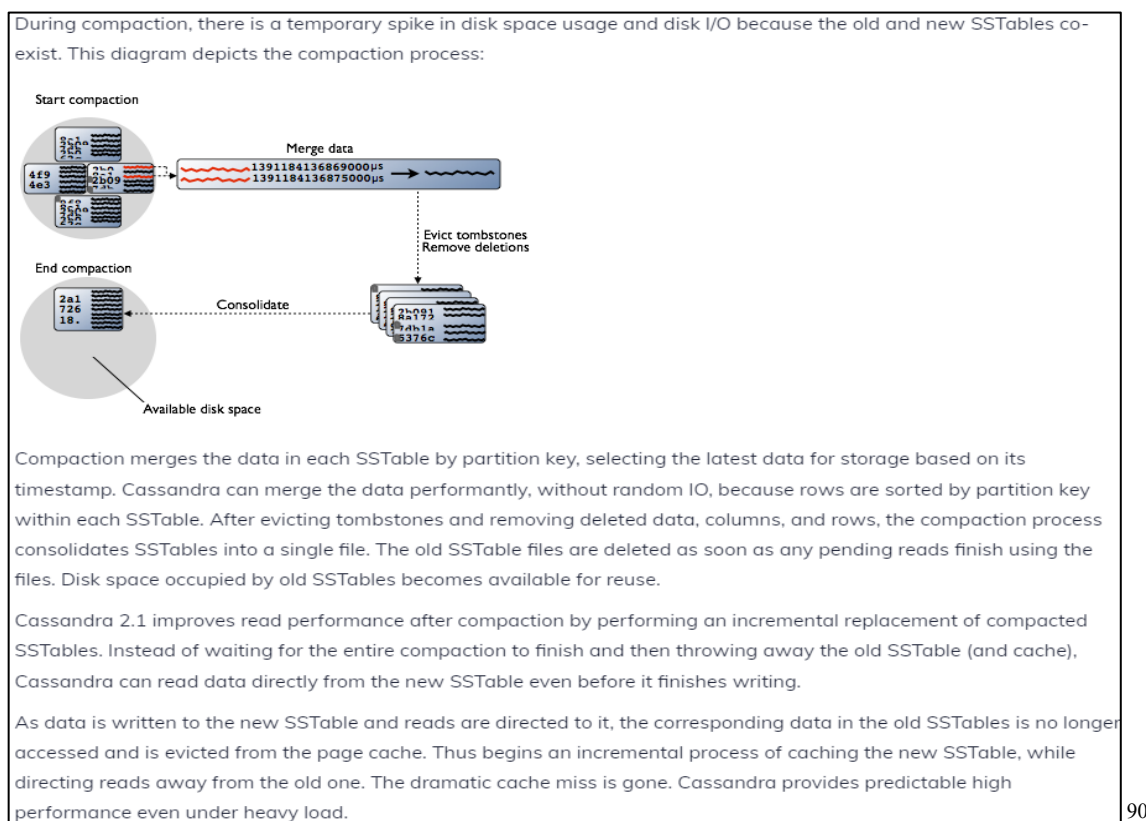
Upper limit of number of SSTables before a compaction is triggered. Not used for `LeveledCompactionStrategy`.

89

⁸⁸ <https://docs.datastax.com/en/cassandra-oss/3.0/cassandra/dml/dmlHowDataWritten.html>

⁸⁹ <https://cassandra.apache.org/doc/stable/cassandra/operating/compaction/#compaction-options>

144. On information and belief, by using Apache Cassandra, Lenovo performs the step of reallocating at least some of said data on said storage device to enhance the performance of said storage device based on said communications. For example, the step of reallocating can occur during minor compactions and can entail merging SSTables. By way of further example, the step of reallocating can also enhance performance of the storage device (e.g., disk) reducing access (“read”) time and making additional disk space available for reuse.



90

145. Lenovo directly infringed the '832 Patent through use, promotional demonstrations, testing, repairs, and instructional guidance of the Kafka Streams using RocksDB Universal Compaction and also systems using Apache Casandra, performing at least the steps of claim 25.

⁹⁰ https://docs.datastax.com/en/cassandra-oss/2.1/cassandra/dml/dml_write_path_c.html

146. Defendants have indirectly infringed one or more claims of the '832 Patent by knowingly and intentionally inducing others, including Lenovo customers and end-users, to directly infringe, either literally or under the doctrine of equivalents, by making, using, offering to sell, selling, and/or importing into the United States products that include the infringing technology.

147. Defendants, with knowledge that these products, or the use thereof, infringed the '832 Patent at least as of January 30, 2024,⁹¹ knowingly and intentionally induced, direct infringement of the '832 Patent by providing these products to end-users for use in an infringing manner. Additionally, on information and belief, Defendants have adopted a policy of not reviewing the patents of others, including specifically those related to Defendants' specific industry, thereby remaining willfully blind to the '832 Patent at least as early as the issuance of the '832 Patent.

148. Defendants have induced infringement by others, including end-users, with the intent to cause infringing acts by others or, in the alternative, with the belief that there was a high probability that others, including end-users, infringe the '832 Patent, but while remaining willfully blind to the infringement. Defendants have induced infringement by their customers and end-users by supplying them with instructions on how to operate the infringing technology in an infringing manner, while also making publicly available information on the infringing technology via Defendants' website, product literature and packaging, and other publications.

149. Plaintiffs have suffered damages as a result of Defendants' direct and indirect infringement of the '832 Patent in an amount to be proven at trial.

⁹¹ Plaintiffs notified Defendant Lenovo Group Limited of its infringement of the '832 Patent in the correspondence on January 30, 2024. Therefore, at least Defendant Lenovo Group Limited was on notice of its infringement of the '832 Patent no later than January 30, 2024.

150. Defendants have committed acts of infringement that Defendants actually knew or should have known constituted an unjustifiably high risk of infringement of at least one valid and enforceable claim of the '832 Patent. Defendants' direct and indirect infringement of the '832 Patent was willful, intentional, deliberate, and/or in conscious disregard of rights under the patent. Plaintiffs are entitled to an award of treble damages, reasonable attorney fees, and costs in bringing this action.

DEMAND FOR JURY TRIAL

Plaintiffs hereby demands a jury for all issues so triable.

PRAYER FOR RELIEF

WHEREFORE, Plaintiffs pray for relief against Defendants as follows:

- a. Entry of judgment declaring that Defendants have directly and/or indirectly infringed one or more of the Patents-in-Suit;
- b. Entry of judgment declaring that Defendants' infringement of one or more of the Patents-in-Suit is willful;
- c. An order pursuant to 35 U.S.C. § 283 permanently enjoining Defendants, their officers, agents, servants, employees, attorneys, and those persons in active concert or participation with them, from further acts of infringement of the '989 Patent, the '930 Patent, the '539 Patent, and the '332 Patent.
- d. An order awarding damages sufficient to compensate Plaintiffs for Defendants' infringement of the Patents-in-Suit, but in no event less than a reasonable royalty, together with interest and costs;
- e. Entry of judgment declaring that this case is exceptional and awarding Valtrus its costs and reasonable attorney fees under 35 U.S.C. § 285; and

f. Such other and further relief as the Court deems just and proper.

Dated: January 27, 2025

Respectfully submitted,

/s/ Vincent J. Rubino, III

Alfred R. Fabricant

NY Bar No. 2219392

Email: ffabricant@fabricantllp.com

Peter Lambrianakos

NY Bar No. 2894392

Email: plambrianakos@fabricantllp.com

Vincent J. Rubino, III

NY Bar No. 4557435

Email: vrubino@fabricantllp.com

FABRICANT LLP

411 Theodore Fremd Avenue

Suite 206 South

Rye, New York 10580

Telephone: (212) 257-5797

Facsimile: (212) 257-5796

***ATTORNEYS FOR PLAINTIFFS
VALTRUS INNOVATIONS LTD. and
KEY PATENT INNOVATIONS LTD.***