



The Anatomy of a Full Path URL

Protocol, Domain, Hostname, Subdomain, Path, and more...

Home / Cybersecurity / The Anatomy of a Full...

Estimated Reading Time: 6 minutes

In a previous blog, we explored the differences between [base domains and full path URLs](#) and their importance in web filtering—particularly how full path URL categorization is critical to [malicious detection](#) and [phishing detection](#) tools that identify and protect users against malicious websites and sources.

In this post, we wanted to take a step back and cover the basics—the individual structural elements of a URL (Uniform Resource Locator). We hope that you'll find this resource helpful—whether as a refresher or as material to add to your ongoing training processes for employees. After all, understanding how a URL is structured is an important step to identifying malicious and unsafe sites.

Table of Contents



Basics: What is a URL?

A [URL, or Uniform Resource Locator](#), is effectively a unique web address. It represents the “location” of a specific resource on the internet. The term URL refers to a subset of [Uniform Resource Indicators \(URIs\)](#) that, in addition to identifying the web resource—also provides the means of locating it.

All webpages are resources online and all have a URL. But not all URLs point to webpages. Every resource on the internet has a URL including: webpages, files, images, media, web applications, services, etc.

In fact, each webpage on the internet is made up of a number of these assets (or resources) including: HTML (Hypertext Markup Language), Hyperlinks (to other pages), CSS (Cascading Style Sheets), Javascript, images, files, code from numerous other programming languages, and more. All of these have their own URLs—with the same structural elements.

Depending on the URL, it may contain some or all of the following:

- [Domain Name](#) (always present)
- [Top Level Domain](#) (TLD) (always present)
- [Subdomain](#)
- [Protocol](#) (always present, not always visible)
- [Path/File](#) (always present, not always visible)
- [HTML Anchors](#)
- [Parameters](#)

So let's jump in.

The Hostname (TLD, Domain Name, Subdomain)

Let's start with the three elements that are the foundation of every URL. Domain Name, Top Level Domain (TLD), and Subdomain.

Top Level Domain

A [top-level domain, or TLD](#), is—as its name suggests—one of the domains at the highest level of the Domain Name System hierarchy. It is the last labelled part in a fully qualified domain name.

Management and responsibility for TLDs is delegated to organizations by the [Internet Corporation for Assigned Names and Numbers \(ICANN\)](#) and the [Internet Assigned Numbers Authority \(IANA\)](#), which maintains the [DNS root zone](#).

Examples of TLDs include:

- .com, .net, .org, .edu



- .us, .uk, .de
- [And more..](#)

Domain Name

A [domain name](#) is the registered identification “string” (or word/phrase) used by the [Domain Name System](#) to define a specific area of control and autonomy (aka a website location). A domain name is a level below the TLD. A single TLD may contain hundreds of thousands and even millions of individual “second-level” domains. Domain may also be referred to as [second-level domain](#).

Examples of domain names are (in bold):

- **zvelo.com**
- **apple.com**
- **amazon.com**
- **facebook.com**
- **google.com**

Subdomain

Subdomains a bit more nuanced—expressing a relative dependence and representing part of a higher level domain. For instance, in **zvelo.com**, **zvelo** is a subdomain of the **.com** domain. As another example, our [zveloLIVE tool](#) is located on the ‘tools’ subdomain of the **zvelo.com** domain. The latter example is the more commonly referenced use of subdomain.

In this way, subdomains are commonly used to divide up domains into smaller segments for communication purposes, content type, internationalization (language translation), or for other reasons.

Some of the most common examples of subdomains include (in bold):

- **www.example.com**
- **blog.example.com**
- **support.example.com**
- **mail.example.com**

NOTE: A URL can exclude or omit the submit domain and the hostname may not include subdomain. This is referred to as a “**naked domain**”. For example, our website does not show a subdomain:

```
https://zvelo.com .
```

Together, these three elements of a URL are the most commonly understood and recognized.

READ: Increase in Mobile Phishing Scams Targeting Apple ID Users



Protocol, Path/File, and More...

Whether you're familiar with them or not, the following URL elements are ubiquitous across the internet and have significant impact on your daily browsing life, privacy, and computer security.

Protocol

The protocol (also referred to as transfer protocol or scheme) in a URL determines how data is transferred between the host and a web browser (or client). HTTP and HTTPS (secure) are two of the most common protocols you'll find in most URLs, though there are other internet protocols such as FTP, DNS, DHCP, IMAP, SMTP, etc.

You'll see the communication protocol (also known as scheme) before any subdomain in a URL. The protocol is followed by a colon and two forward slashes separating it from the hostname.

For example: `https://zvelo.com`

Path/File

The path refers to the exact location of a page, post, file, or other asset. It is often analogous to the underlying file structure of the website. The path resides after the hostname and is separated by "/" (forward slash). The path/file also consists of any asset file extension, such as images (.jpg or .png, etc.), documents (.pdf or .docx), and more.

But not all URLs will display a path.

For instance when you visit the homepage of a site on many modern website, you may not see a path or file name. This is because many modern websites can "rewrite" URLs (like the homepage) for simplicity and elegance—such as omitting the typical "index.html".

For example: `https://www.example.com/blog/category/individual-article-name/` or
`https://example.com/media/upload/filename.jpg`

In some cases, websites use a naming structure that incorporates the date (month/day/year) or categories, or other separators to organize content or just the URL itself.

HTML Anchors

HTML Anchors (or fragments) are used on websites to implement “bookmarks” and internal page navigation elements. These can be used to provide links to specific locations within a page. Anchors will be located immediately after the file/path of a URL (when present).

For example, here is an internally linked URL with an HTML anchor that would take you to the next section:

`https://zvelo.com/anatomy-of-full-path-url-hostname-protocol-path-more#parameters` ([Try it](#))

Parameters

Parameters are found at the very end of the URL or within the path, depending on the implementation. URL Parameters are represented in key/value pairs, beginning with a ‘?’ and separated by an ampersand ‘&’. They can also be dynamically set in the path as values separated by slashes and other characters (depending on the system being used and how it’s implemented). Parameters are commonly used for tracking and analytics—as well as encoding specific information for use within websites and applications.

For example, a URL with parameters:

- `https://www.example.com/solutions?user=123&color=blue`
- `https://www.example.com/solutions/user/123/color/blue`

Alternatively, you might find a URL with a Google campaign tracking parameter such as:

- `https://www.example.com/?utm_source=google&utm_medium=banner&utm_campaign=urlfilter`

Other Considerations

These URL elements instruct your web browser how to connect to a web host, acquire resources, interact with back-end systems, and display web content for you. Knowing what to look for, particularly with transfer protocol and hostname, can help web users spot and identify scams and malicious websites before falling victim.

With [web filtering](#) in mind, most “basic” and “consumer” solutions do not support [full-path URL categorization](#)—and therefore do not account for path/file, parameters, or protocol. These offerings rely on [DNS filtering](#) to protect users and network traffic from accessing “undesirable” websites. This is limiting because it only allows for classification(s) or [objectionable/malicious](#) identifiers to be assigned at the site level (or base domain, aka hostname).

READ: Evaluating a URL Database and Classification Technologies for DNS and Web Filtering

What happens when a site is compromised and that otherwise “clean”, high-traffic site begins to disseminate a [malicious file](#) from a single source or page? For that, check out our blog on the [differences between base domain and full Path URLs](#). Or check out our other [solutions](#) for [Cyber Threat Intelligence and Network Security](#).

BELOW ARE A COUPLE OF ARTICLES TO HELP YOU LEARN MORE ABOUT WHY IT’S CRUCIAL FOR MALICIOUS DETECTION AND CYBER THREAT INTELLIGENCE SOLUTIONS TO HAVE VISIBILITY AND BLOCKING CAPABILITIES ON FULL-PATH URLS TO ENSURE YOU ACHIEVE MAXIMUM PROTECTION AS WELL AS VALUE.

- [In-House Cyber Threat Intelligence Begs a Multi-Million Dollar Question](#)
- [Whitelisted Sites Are Being Used to Deliver Malware](#)
- [Cyber Threat Intelligence and Network Security](#)

We hope you’ve found this helpful! Now, go forth and browse safely!

Categories: Cybersecurity, DNS Filtering, Internet Safety & Privacy, Malicious Detection, Technology & Insights, URL Database, Web Filtering & Parental Controls •

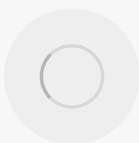
By zvelo • November 2, 2018

Tags: [Base Domain](#) [Cyber Threat Intelligence \(CTI\)](#) [Cybersecurity](#) [DNS Filtering](#) [Full Path URL](#) [Indicators of Compromise \(IOCs\)](#)

[Malicious Detection](#) [Malicious URLs](#) [SASE](#) [Top Level Domain](#) [TTPs: Tactics Techniques and Procedures](#) [URL Analysis](#) [URL Classification](#)

[URL Filtering](#)

Share this post



Author: zvelo

<https://zvelo.com/about/>

Making the Internet Safer and More Secure. zvelo provides industry-leading cyber threat intelligence and URL classification data services. zvelo's proprietary AI-based threat detection and categorization technologies, combines curated domains, threat and other data feeds, with the clickstream traffic from its global partner network of 1 billion users and endpoints to provide unmatched visibility, coverage, reach and accuracy. zvelo powers applications and solutions for the world's leading providers of web filtering, endpoint detection and

response (EDR), extended detection and response (XDR), Secure Access Service Edge (SASE), brand safety and contextual targeting, cyber threat intelligence platforms, threat analysis, and more.

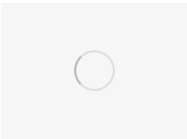
PREVIOUS

< **Data Piracy Challenges to DaaS Business Models**

NEXT

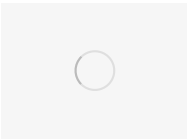
What is an IDN Homograph Attack and How Do You Protect Yourself? >

Related Posts



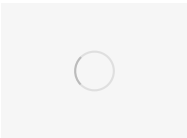
Tycoon 2FA: Protect Against Phishing Kits That Bypass MFA

July 23, 2024



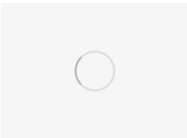
Threat Advisory: Polyfill.io Supply Chain Attack

July 2, 2024



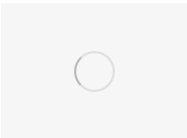
Deepfakes: Escalating Threats and Countermeasures

April 24, 2024



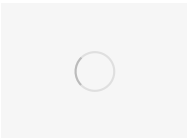
Phishing Awareness Training for the Generative AI Era

March 13, 2024



Malicious AI: The Rise of Dark LLMs

February 8, 2024



Embracing Curated Threat Intelligence in Times of Cost-Cutting

December 14, 2023

What Are You Looking For?

Type and hit enter ... 

Browse By Category

> Ad Tech & Brand Safety

> AI & Machine Learning

- > [Announcements, News & PR](#)
- > [Cybersecurity](#)
- > [Threat Intelligence](#)
- > [Internet Safety & Privacy](#)
- > [IoT Security Technology](#)
- > [Malicious Detection](#)
- > [Mobile & Subscriber Analytics](#)
- > [Phishing Detection](#)
- > [Technology & Insights](#)
- > [URL Database](#)
- > [Web Filtering & Parental Controls](#)

Recent Blogs

- Tycoon 2FA: Protect Against Phishing Kits That Bypass MFA July 23, 2024
- Threat Advisory: Polyfill.io Supply Chain Attack July 2, 2024
- Deepfakes: Escalating Threats and Countermeasures April 24, 2024
- Phishing Awareness Training for the Generative AI Era March 13, 2024

Datasheets & Reports

- > [Cost Analysis | zvelo Threat Intel vs OSINT](#)
- > [zveloCTI™ | Cyber Threat Intelligence](#)
- > [PhishBlocklist™ | Phishing Intelligence](#)
- > [PhishScan™ | Real-Time Phishing Threat Verification](#)
- > [Malicious Detailed Detection Feed™ | Malicious Intelligence](#)
- > [zveloDB™ | URL Database](#)
- > [zveloCAT™ | Real-Time Web Classification Engine](#)

> Report | Emotet's Return Foreshadows Ransomware Surge

> White Paper | 6 Steps to Drive Value from Cyber Threat Intelligence

Cyber Threat Definitions





+1 (720) 897-8113
+1 (877) 77-zvelo
(Toll Free)

zvelo US
Headquarters
Denver, Colorado
USA

CONTACT US

Products	Use Cases	Resources	Company
zveloDB™ URL Database	(DRPS) Digital Risk Protection Services	Blog, News & Press	About zvelo
zveloCAT™	(XDR) Extended Detection and Response	Case Studies	Contact Us
zveloCTI™ Cyber Threat Intelligence	(SASE) Secure Access Service Edge	Datasheets, White Papers, & Reports	Careers at zvelo
Malicious Detailed Detection Feed™	(MSSPs) Managed Security Service Providers	Videos, Webinars, & Virtual Events	Executive Leadership
PhishBlocklist™	(OT) Operational Technology Security	Cyber Threat Definitions	
PhishScan™	Threat Research		
	Web & DNS Filtering		
	Brand Safety & Contextual Targeting		

PrivacyLegal

© 2024 zvelo, Inc. All rights reserved.