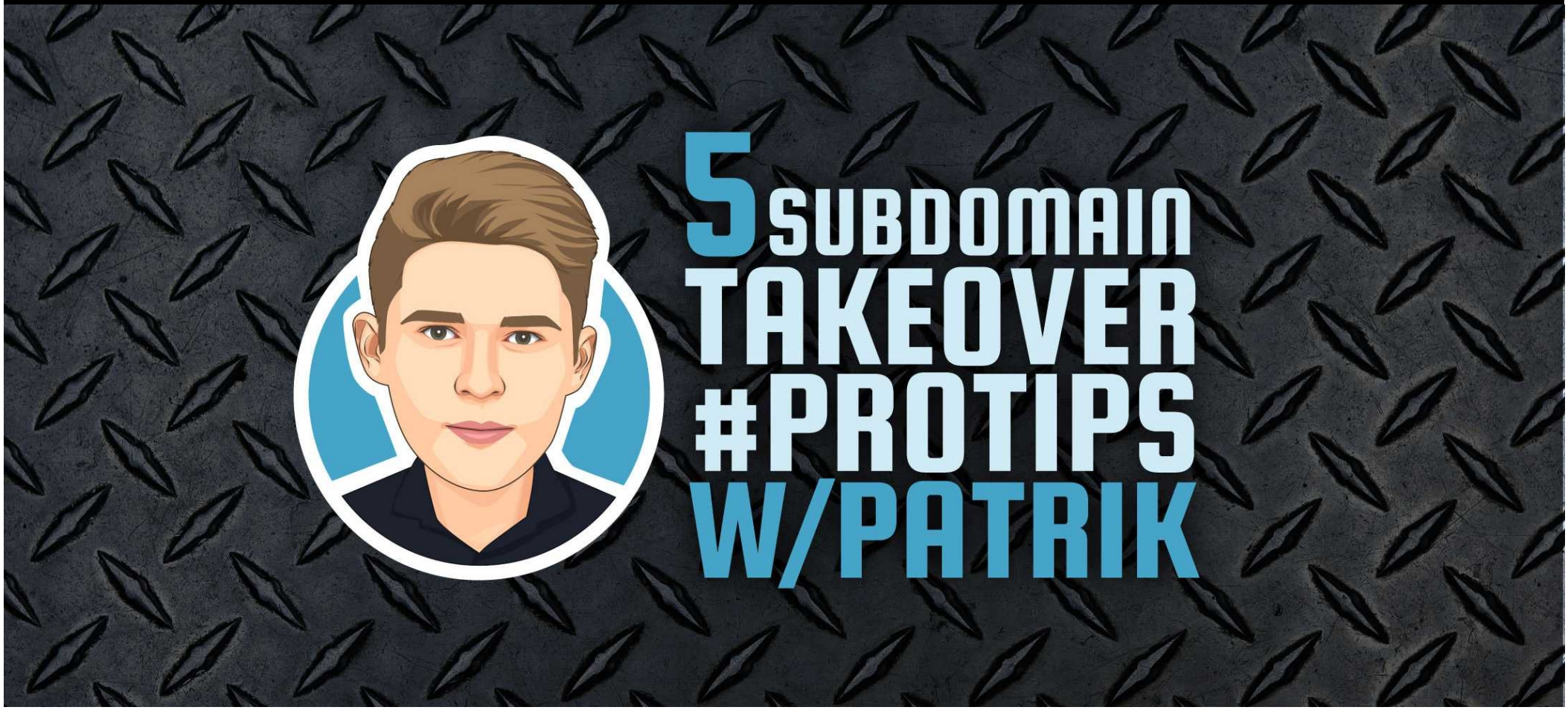


TIPS TOOLS RECONNAISSANCE

SECURITYTRAILS BLOG · OCT 01 · SECURITYTRAILS TEAM

5 Subdomain Takeover #ProTips

Reading time: 7 minutes

Facebook Twitter LinkedIn

We at SecurityTrails are happy to introduce a new blog series where we'll be sharing tips and step-by-step research processes from the best security researchers around. You'll get a glimpse into their methodology, the tools they use and much more, with ProTips!

SEARCH BLOG

Search...

TABLE OF CONTENTS

Today we're speaking with Patrik Hudak, an accomplished security researcher and bug bounty hunter. Patrik was

already featured in our interview series when we touched upon his research on subdomain takeovers, but now we'll go deeper into his process. He'll share his ProTips and methodology behind finding vulnerable subdomains.

Before we dive in, you can check out Patrik's post on the basics of subdomain takeovers, their implications and the most common scenarios of attacks exploiting this particular vulnerability.

#ProTips

1. Finding services that are currently vulnerable to takeovers
2. Patrik's starred Github projects
3. Advanced Recon — Azure
4. Building better wordlists
5. Not all subdomains are created equal

#ProTip I: Know the services that are vulnerable to takeovers

There has been a rapid improvement over the past year—many of the providers that were vulnerable to subdomain takeovers started to implement mitigations strategies. But even with many false positives, there are still some cases where subdomain takeover exists. Personally, I see domains using Microsoft Azure in CNAME records as being the most vulnerable.

Subdomains vulnerable to subdomain takeovers in 2019

This list is updated regularly so you can check it out on GitHub to see which service are still vulnerable.

For a more detailed list of specific subdomains that might be vulnerable, check this link.

#ProTip 2: Check out Patrik's starred Github projects

tar

massdns C

Created by blechschmidt

Star

A high-performance DNS stub resolver for bulk lookups and reconnaissance (subdomain enumeration)

166 FORKS **1.1K** STARS

commonspeak2

Created by assetnote

Leverages publicly availa
content discovery and su

24 FORKS **113** STARS

tar

Amass Go

Created by OWASP

Star

In-depth Attack Surface Mapping and Asset Discovery
www.owasp.org/index.php/OWASP_Amass_Project

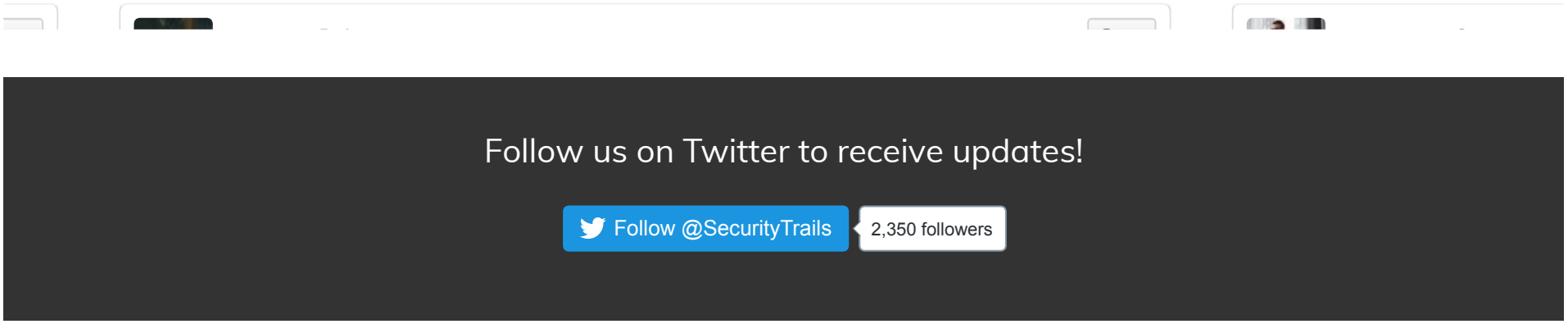
289 FORKS **2.0K** STARS

can-i-take-over-x

Created by EdOverflo

"Can I take over XYZ?" –
with dangling DNS recor

225 FORKS **1.3K** STARS



#ProTip 3: Perform advanced recon – Azure

Microsoft Azure has many services that generate a unique IP address and domain name, and these domains are often used in CNAME records. Subdomain takeover arises when the resource is removed from the Azure portal and DNS zone is kept intact. The verification is fairly simple: if the subdomain of one of Azure's services responds with NXDOMAIN for DNS requests, there is a high chance that the takeover is possible. I say "high chance" because there are a few edge cases (e.g. disabled resource), in which NXDOMAIN is returned, however, the takeover is not possible.

Over the last year, I reported 9 subdomain takeover reports and 9 of them were tied to Microsoft Azure. To boost your chances of a successful report, I suggest you go through each of the Microsoft Azure services and try to create a new

instance. It can be Cloud Application, Virtual Machine, Load Balancer, etc. If you do that, you will clearly see that each of those services has a different root domain. Personally, I currently have 17 domains that Microsoft Azure is doing (*.visualstudio.com,* .azurecr.io, ...). This type of information gives you a huge advantage over other hackers who don't know what to search for.

#ProTip 4: Building better wordlists

To find subdomain takeovers, you first need a list of active domains. To get one, you can use open-source tools such as Amass. Over the last year, I realized that there are still many more active domains you can find on the Internet. I always hated the idea of brute forcing, but balancing the pros and cons, I decided to start doing "smart brute forcing". This means creating a wordlist based on your targets. I mostly use the Commonspeak2 wordlist to generate the first batch of brute force possibilities.

Commonspeak2 is the brainchild of Assetnote's creators and it parses through datasets from Google's BigQuery. In querying those datasets, it generates wordlists that are updated regularly to reflect the newest technologies on the

Internet. Another thing that sets it apart from other wordlists you can find online is that it extracts datasets from Stack Overflow and HackerNews, the biggest communities of programmers, developers and other IT folk online, so it accesses the most current files, directories and subdomains.

My main focus, however, is using already-found domains to generate other possible alteration. See, when there is an active domain called “app.staging.example.com”, there is a high chance that there is a “sibling” domain named “app.production.example.com”. Tools such as Amass might not find it, so you need to deal with it on your own. For this purpose, I recommend using my new tool [dnsgen][dnsgen] to generate brute force possibilities. Dnsgen is a tool that generates a combination of domain names from provided input and the combinations are created using wordlists. This is done with the following technique and let’s say that the wordlist contains only the word stage:

- **Insert word on every index** — Creates new subdomain levels by inserting the words between existing levels. foo.example.com → stage.foo.example.com, foo.stage.example.com
- **Insert num on every index** — Creates new subdomain levels by inserting the numbers between existing levels. foo.bar.example.com → 1.foo.bar.example.com, foo.1.bar.example.com, 01.foo.bar.example.com, …
- **Increase/Decrease num found** — (In development) If number is found in an existing subdomain, increase/decrease this number without any other alteration. foo01.example.com → foo02.example.com, foo03.example.com, …
- **Prepend word on every index** — On every subdomain level, prepend existing content with WORD and WORD-. foo.example.com → stagefoo.example.com, stage-foo.example.com
- **Append word on every index** — On every subdomain level, append existing content with WORD and WORD-. foo.example.com → foostage.example.com, foo-stage.example.com

- **Replace the word with word** — If word longer than 3 is found in an existing subdomain, replace it with other words from the wordlist. (If we have more words than one in our wordlist). stage.foo.example.com → otherword.foo.example.com, anotherword.foo.example.com, ...
- **Extract custom words** — Extend the wordlist based on target's domain naming conventions. Such words are either whole subdomain levels, or - is used for a split on some subdomain level. For instance mapp1-current.datastream.example.com has mapp1, current, datastream words. To prevent the overflow, user-defined word length is used for word extraction. The default value is set to **6**. This means that only words strictly longer than **5** characters are included (from the previous example, mapp1 does not satisfy this condition).

Then, push all of it to massdns to get back only the active ones. You can find more info on my blog.

#ProTip 5: Not all subdomains are created equal

When looking for subdomain takeovers, one of the primary things that gives you an edge is finding more subdomains for the target. However, there are still other people doing the same. To be always ahead of the curve, you need:

- Fast periodic checks to reveal subdomain takeovers when DNS records are changed
- Fast subdomain enumeration to find new domains once they are created

And subdomain takeover might only be available for a small amount of time. You need scripts to automate the whole workflow for you. Ideally, these scripts are able to register the resource for you, blocking other hunters from acquiring them. This might sound complex—however, Azure API allows you to do it pretty easily.

We hope you've enjoyed learning from Patrik in this first entry in our #ProTips series. We'll continue bringing you experts who'll share deep insights into their research, so please let us know who you'd like to see next, by emailing us at hello@securitytrails.com. Stay tuned to our blog for new additions and more great information!

‹ PREVIOUS NEXT ›

Related Posts

What's My DNS? How to Find Domain DNS Records with our DNS Checker

Exploring the best ways to answer the question: What's my DNS?

Cybersecurity Red Team Versus Blue Team — Main Differences Explained

We've previously explored the Top 20 OSINT Tools available, and today we'll go through the list of top-used Kali Linux software.

Cybersecurity Fingerprinting Techniques and OS-Network Fingerprint Tools

We explore what a fingerprint is in cyber security, different types of fingerprint techniques, and some of the most popular fingerprinting tools in use.

- NEW DNS Enumeration: Top DNS Recon Tools and Techniques
- 5 Subdomain Takeover #ProTips
- CMS Detector: What CMS a Website is Using and the Best Tools to Find ...
- ASN Lookup Tools, Strategies and Techniques
- What is Reverse DNS? Top Tools for Performing a Reverse DNS Lookup