Deanonymization of TOR HTTP Hidden Services

Ionut Cernica
Table Of Contents

- Introduction
- Known deanonymization techniques
- Downgrading the protocol
- Other techniques developed by me
- How to fix
- Demo time and Closing Remarks
The Onion Router

The Internet

The Dark Net

http://*.onion
Hidden Service

Allow users to publish their service without revealing their identity

Configuration

`HiddenServiceDir /var/lib/tor/hidden_service/`
`HiddenServicePort 80 127.0.0.1:80`

Hidden service address

mhphb7utr2eqmul5mgggk5apf6dyjlldgujfvxohjiqu5pd2b7scg6qd.onion
Known Deanonymization Techniques

- /server-status
- Key Certificate
- Search for the onion address
- Gzip
- favicon.ico
- Hack the server
http://*.onion/server-status
# Key Certificate

## Public Key Info

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Elliptic Curve</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key Size</td>
<td>256</td>
</tr>
<tr>
<td>Curve</td>
<td>P-256</td>
</tr>
</tbody>
</table>

## Miscellaneous

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Signature Algorithm</td>
<td>SHA-256 with RSA Encryption</td>
</tr>
<tr>
<td>Version</td>
<td>3</td>
</tr>
<tr>
<td>Download</td>
<td>PEM (cert) PEM (chain)</td>
</tr>
</tbody>
</table>
Key Certificate

• TLS certificate might be indexed on the surface web and can lead to the same resource from the dark net or other services of the same actor

• Shodan indexes information from the internet including TLS information
Search for the onion address

- Very little chance of success
- Just search the onion address on search engines like Google, Bing, DuckDuckGo or Shodan
Gzip Compression

- Jose Carlos found that around 10% of the webservers leak the remote date when compressing HTTP Responses with gzip.
- It's not a problem in TOR and it's not a bug in the protocol as well and is not a problem with the GZIP.
- It helps you get an idea of where one or another server is hosted.
• favicon is that tiny icon that users see in the browser's URL bar

• It is possible to match favicons found on dark web with favicons on the internet using Shodan

• The Quantum ransomware group is an example:

Using its favicon from the dark web, Talos found its equivalent on the surface web and could locate the threat actor’s web server
Hack the server

Find vulnerabilities in order to deanonymize the server

- Remote Command Execution
- Local File Inclusion

Make a research on the server and look for other vulnerabilities that can leak the servers IP
Downgrading The HTTP Protocol

- How did I find this?
  - Strange requests in my access.log file using HTTP/1.0 protocol:
    "GET /NotFoundNotFoundNotFoundNotFoundNotFoundNotFoundNotFoundNotFoundNotFou......... HTTP/1.0" 400 802 "-
    "masscan - for more info go - http://something.org"
    
  - Same behavior on my honeypots
  - The requests keep coming for days
  - I dumped the hole request
  - I replicated and got an internal virtual host of mine that I didn’t want to be public
  - This was the moment when I realized the potential of this flaw, both on the internet and dark web
  
  - Started to develop an improved version of this /NoTFound…. request in order to exfiltrate the IP or unknown domains
Downgrading The HTTP Protocol

• **Why this behavior?**
  - Is not because of a security problem in apache-based servers like: apache2, nginx or tomcat
  - Is all about configuration
  - The server must choose one of the domains to forward the request
  - The client doesn’t supply a “**Host: example.com**“ header and to do that we can choose the first version of HTTP protocol -> HTTP/1.0
  - The server will choose the first virtualhost
  - In the response we can find first declared virtualhost. Which might be a domain or an IP or just localhost
The leak

• The leak is in the:
  - Triggered exceptions
    - Doesn’t work on all apache-based servers
    - HTTP/1.0 400 Bad Request
    - HTTP/1.0 403 Forbidden
    - Even the 404 NotFound sometimes discloses the IP
  - Server redirects
    - The best way to leak the IP
    - works in servers like nginx, apache2, tomcat
Don’t forget to downgrade
Don’t forget to downgrade
HTTP/1.0 400 Bad Request – Long Header
HTTP/1.0 400 Bad Request

- The same result with a long URL or a file upload that exceeds the server limit
HTTP/1.0 403 Forbidden

- Very easy to trigger
- I used the ^\.ht* rule and you can visit /\.html to not attract attention
- The rule above is for .ht files, like .htaccess or .htpasswd, etc
Server redirects

• The best way to trigger redirect is to search for directories that serves static files
  - servers based on apache must put a / at the end

  - Exampe: if we visit http://server.com/dir it will auto redirect to http://server.com/dir/

  - most cases you can find on the first page directories

  - if there is an app without directories that serves something, then try some predefined directories that exists in some servers like “/img”, “/css”, “/icons”, “/js”

  - this directories are available only if you visit with the IP 127.0.0.1, which most of the time in TOR this is the IP that you access the application
Trigger the redirect

```html
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="/css/styles.css">
</head>
<body>
  ...
</body>
</html>
```
<!DOCTYPE html>
<html>
<head>
    <link rel="stylesheet" href="/css/styles.css">
</head>
<body>
...
</body>
</html>
Trigger the redirect and leak the IP
Apply this on the Internet

• I didn’t make a research on this. I suppose is at the limit of legality

• It appears that others did this before, but I don't know if they followed this problem

• I expect to exfiltrate domains that normally you cannot get from an IP. It's a kind of reverse DNS on a specific IP
Other techniques - ETag

- The ETag or entity tag is part of HTTP, the protocol for the World Wide Web
- Downgrade the protocol has nothing to do with this one
- It suppose to take the ETag of the default domain and search it on the internet
- We can find ETag in the response header
- This can be done if the first page is static and not dynamic
- We can use Shodan
- The chance of success is quite small, but it's worth a try
## Example of ETag

### Request

<table>
<thead>
<tr>
<th>Pretty</th>
<th>Raw</th>
<th>Hex</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /</td>
<td>HTTP/1.0</td>
</tr>
<tr>
<td>2</td>
<td>User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:102.0) Gecko/20100101 Firefox/102.0</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Accept: text/html, application/xhtml+xml, application/xml;q=0.9, image/avif, image/webp, <em>/</em>; q=0.8</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Accept-Language: en-US, en; q=0.5</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Accept-Encoding: gzip, deflate</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Connection: close</td>
<td></td>
</tr>
</tbody>
</table>

### Response

<table>
<thead>
<tr>
<th>Pretty</th>
<th>Raw</th>
<th>Hex</th>
<th>Render</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>HTTP/1.1 200 OK</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Date: Thu, 28 Jul 2022 08:31:34 GMT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Server: Apache/2.4.52 (Ubuntu)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Last-Modified: Thu, 07 Jul 2022 09:01:20 GMT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>ETag: &quot;5-5e3335441f800&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Accept-Ranges: bytes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Content-Length: 5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Connection: close</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Hello</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Other techniques - Same Network Technique

• This technique involves domain enumeration on a given host

• Because the number of known .onion address on TOR is small, then it is doable to search on the same server other TOR domains

• If we find multiple domains on a single server then is enough to find the IP address of one of the domains to find the others

• I remember I found a hosting provider on TOR for hidden services, but the owner didn’t used a good sandbox and I was able to find the IP address. Then if I would make a domain enumeration on that server I would find the other hidden services hosted on that server

• To achieve that we need to change in “Host:” header the name of the hidden service with the ones from the list, one at a time and look for the response if is the same
KEEP CALM
IT'S DEMO TIME
Closing Remarks

- We saw how many possibilities are there to trigger an exception in order to leak an IP or a domain in a server. If you are lucky enough, the IP can be 127.0.0.1 or a domain that doesn’t specify something.

- Also, how easy is to trigger a redirect in order to leak the IP.

- Keep in mind to create a virtual host for localhost or 127.0.0.1 to be sure that all the traffic for http/1.0 will go through it.

- Try not to reuse certificates from other projects.

- Disable server-status page.

- All of this are configuration problems and is important that any sysadmin to know about this.
Thanks

Do you have any questions?

Ionut.cernica@gmail.com