



HydrantID SSL Discovery Utility

June 15, 2015
Version 1.0

Introduction

The SSL Discovery Utility is a command line tool that uses a TLS/SSL connection to interrogate a range of IPv4 network addresses and ports to identify any SSL certificates used. The output can either be to a terminal window, or to an Microsoft Excel file to save and categorize the results.

Disclaimer

Copyright (c) 2015, Avalanche Cloud Computing d.b.a HydrantID

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Usage

The tool is programmed in Node.js (<http://nodejs.org>) and is run from the command line as follows:

```
Discover SSL certificates within a network.
```

```
Usage: ssl_discovery -n <cidr network | hostname | filename>
```

```
Examples:
```

```
ssl_discovery -n 192.168.1.0/24 -o localnetwork.xlsx -v  
Scan network 192.168.1.1-192.168.1.254
```

```
ssl_discovery -n filename.txt  
Scan the CIDR, IP or hosts listed in filename.txt
```

```
ssl_discovery -n www.example.com  
Scan www.example.com host
```

```
ssl_discovery -n www.example.com -p 8080 -p 8443  
Scan www.example.com on ports 8080 & 8443
```

```
Options:
```

```
-n, --net          CIDR network or hostname to scan, or file containing IP or  
                  hostnames [required]  
-p, --port        Specific port to probe  
-o, --out         Output Excel file (.xlsx)  
-v, --verbose     Verbose output of each ip:port combination
```

The IPv4 address range is specified using CIDR notation (see http://en.wikipedia.org/wiki/Classless_Inter-Domain_Routing#CIDR_notation). A single hostname can be specified instead of a CIDR address.

The output Excel file is optional. If not specified, the results will be output to the screen.

In addition, if the default ports listed in the config.json do not wish to be used, specific ports can be listed with the --port option.

For example, the following will scan the class C address range from 192.168.1.1 to 192.168.1.255 and save the output to the file 'localnetwork.xlsx':

```
$ ssl_discovery -n 192.168.1.0/24 -o localnetwork.xlsx
```

Configuration

Changing the config.json file can set the following parameters:

Parameter	Value
socketTimeout	# of milliseconds to wait for socket to timeout
ports	Comma separated list of ports that are checked on each address for SSL certificates
concurrentHosts*	Number of hosts that are checked simultaneously
concurrentPorts*	Number of ports that are checked simultaneously for each host

The concurrentHosts & concurrentPorts parameters may need to be changed depending on the limit of file descriptors for each process on your system. The value of concurrentHosts*concurrentPorts should be lower than the maximums set for your system (defaults are: MacOS=256, Ubuntu/Linux=1024, Windows=512)

Installation

To install the SSL Discovery tools, perform the following:

1. Install Node.js for your platform from <http://nodejs.org> - make sure to add the "node" tool to your local PATH
2. Use the npm package manager to install the packaged installer with `npm install -g ssl_discovery-0.0.4.tgz`. If permissions errors are received, execute the npm command with superuser/administrator permissions.
3. Run `ssl_discovery -n`

Uninstall

To uninstall the SSL Discovery tools, perform the following:

1. `npm uninstall -g ssl_discovery` - if you receive permission errors, execute the npm command with superuser/administrative permissions.