

Osquery explained

a deep dive in endpoint monitoring:
Understand and plan Osquery
deploy on global scale

Thinks i'm going to talk about

EDR - endpoint detection & response:

how to build a ecosystems for mass incident detection and & response

Thinks i don't talk about

Product pitch

What you can find on the osquery doc pages

How cool is the company I work for

What problem are we trying to solve?

- Too many OS centric products to solve too narrow of a use-case
- New use-case? new vendor
- We are not anymore in a windows centric or single distro world
- Many vendor specific languages
- Too much data to collect, store and analyse

Which kind of people you deal with?

- Do they escalate to IR for further investigation?
- Can your IR start investigation without a confirmed incident?
- Will this overload your IR?

Osquery 101

SQL powered operating system instrumentation, monitoring and analytics.

Released **under** apache license.

Windows, macOS, CentOS, FreeBSD, and almost every Linux OS released since 2011 are supported with no dependencies.

Can be used for DEVOPS compliance and Security.



Tip

[http:// osquery .io/](http://osquery.io/)

[https:// github.com/ face book/ osquery](https://github.com/facebook/osquery)

[https:// osquery.readthe docs.io/ en/ stable/](https://osquery.readthedocs.io/en/stable/)

What it look like?

```
/etc/osquery/  
/usr/share/osquery/osquery.example.conf  
/usr/share/osquery/lenses/{*}.aug  
/usr/share/osquery/packs/{*}.conf  
/var/log/osquery/  
/usr/lib/osquery/  
/usr/bin/osqueryctl  
/usr/bin/osqueryd  
/usr/bin/osqueryi
```

Using it : Osqueryi

```
λ osqueryi.exe
Using a virtual database. Need help, type '.help'
osquery> .tables
=> appcompat_shims
=> arp_cache
=> authenticode
=> autoexec
=> carbon_black_info
=> carves
```

- to list all tables: `.tables`
- to list the schema (columns, types) of a specific table: `.schema table_name` or `pragma table_info(table_name)`; for more details
- to list all available commands: `.help`
- to exit the console: `.exit` or `^D`

Tip

[https:// osquer
y.readthedocs.i
o/ en/ stable/ i
ntroduction/ sq
l/](https://osquery.readthedocs.io/en/stable/introduction/sql/)

[https:// github.com/ fac
ebook/ osquery/ relea
se](https://github.com/facebook/osquery/releases)

Let's play queries!

osquery table magic

All Platforms

acpi_tables
arp_cache
authorized_keys
blacklist
block_devices
chrome_extensions
cpuid
crontab
device_file
device_hash
device_partitions
disk_encryption
etc_hosts
etc_protocols
etc_services
example
file_events
firefox_addons
groups
hardware_events
interface_addresses
interface_details
kernel_info
known_hosts
last
listening_ports
logged_in_users
magic
mounts
opera_extensions

All Platforms

os_version
pci_devices
platform_info
process_envs
process_events
process_memory_map
process_open_files
process_open_sockets
processes
routes
shell_history
smbios_tables
suid_bin
system_controls
system_info
uptime
usb_devices
user_groups
users
yara
yara_events

Darwin (OS X)

ad_config
alf
alf_exceptions
alf_explicit_auths
alf_services
app_schemesapps
authorizations
authorization_mechanisms
browser_plugins
certificates
disk_events
extended_attributes
homebrew_packages
iokit_devicetree
iokit_registry
kernel_extensions
keychain_acls
keychain_items
launchd
launchd_overrides
managed_policies
nfs_shares
nvram
package_bom
package_receipts
preferences
process_file_events
safari_extensions
sandboxes
signature
sip_config
smc_keys
startup_items
temperatures
wifi_networks
xprotect_entries
xprotect_meta
xprotect_reports

Ubuntu, CentOS

iptables
kernel_integrity
kernel_modules
memory_map
msr
shared_memory
socket_events
user_events
apt_sources
deb_packages
rpm_package_files
rpm_packages

Utility

file
hash
osquery_events
osquery_extensions
osquery_flags
osquery_info
osquery_packs
osquery_registry
osquery_schedule
time





Tip

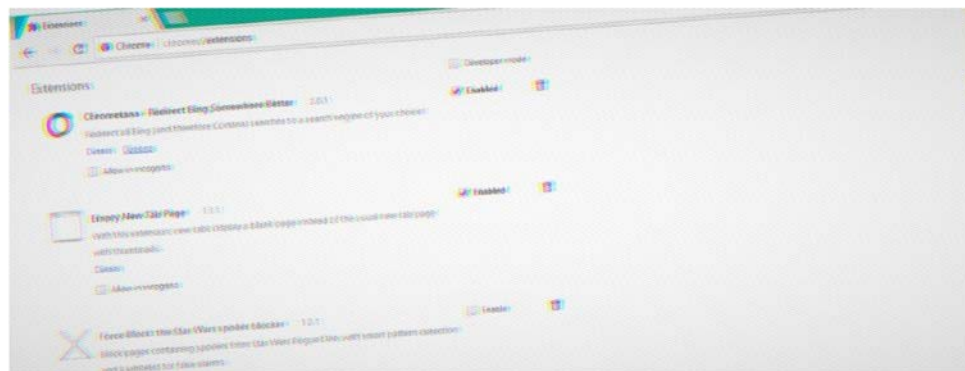
<https://www.bleepingcomputer.com/news/security/eight-chrome-extensions-hijacked-to-deliver-malicious-code-to-4-8-million-users/>

Home » News » Security » Eight Chrome Extensions Hijacked to Deliver Malicious Code to 4.8 Million Users

Eight Chrome Extensions Hijacked to Deliver Malicious Code to 4.8 Million Users

By Catalin Cimpanu

August 16, 2017 12:30 AM 0



Six more developers have had their Chrome extensions hijacked in the past four months, according to

BetternetVPN

```
SELECT * FROM users JOIN chrome_extensions USING  
(uid) WHERE identifier='gjknjjomckknofjidppipffbpoekiipm';",
```

Chrometana

```
SELECT * FROM users JOIN chrome_extensions USING  
(uid) WHERE  
identifier='kaicbfmipfpfpjmlbpejaoaflfdnabnc';",
```

...

Tip

<https://security.googleblog.com/2017/10/behind-masq-yet-more-dns-and-dhcp.html>

Behind the Masq: Yet more DNS, and DHCP, vulnerabilities

October 2, 2017

Posted by Fermin J. Serna, Staff Software Engineer, Matt Linton, Senior Security Engineer and Kevin Stadmeyer, Technical Program Manager

Our team has previously posted about [DNS vulnerabilities and exploits](#). Lately, we've been busy reviewing the security of another DNS software package: [Dnsmasq](#). We are writing this to disclose the issues we found and to publicize the patches in an effort to increase their uptake.

Dnsmasq provides functionality for serving DNS, DHCP, router advertisements and network boot. This software is commonly installed in systems as varied as desktop Linux distributions (like Ubuntu), home routers, and IoT devices. Dnsmasq is widely used both on the open [internet](#) and internally in private networks.

Find dnsmasq installed via Homebrew or MacPorts by enumerating related launchd plist:

```
SELECT * FROM launchd WHERE name LIKE '%dnsmasq%';
```

Find running Docker containers with dnsmasq in the name:

```
SELECT name FROM docker_containers WHERE name LIKE '%dnsmasq%';
```

Discover hosts that are have dnsmasq listening on localhost port 53:

```
SELECT DISTINCT(processes.name), process_open_sockets.local_port FROM processes JOIN  
process_open_sockets USING (pid) WHERE local_port=53 AND processes.name='dnsmasq';
```

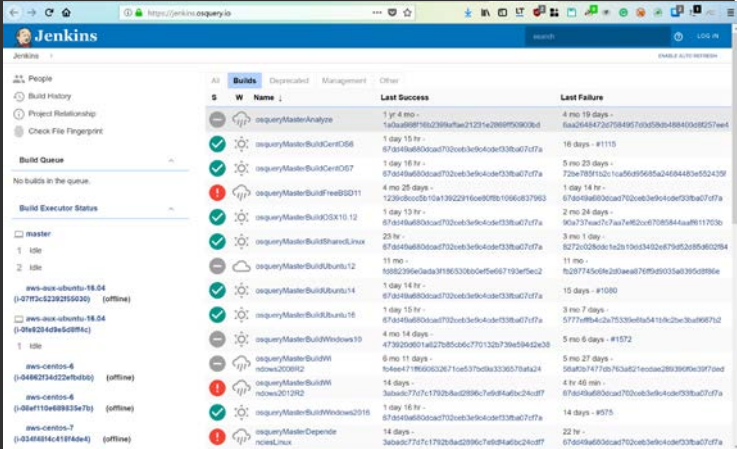
Find users who installed dnsmasq via Homebrew:

```
SELECT * FROM homebrew_packages WHERE name='dnsmasq';
```

Build and test osquery for different platform

For Windows: make_win64_binaries.bat
For Linux and mac: \$make package &&
make_osx_package.sh

Automate with Jenkins!



Tip

[https:// osquery.readthedocs.io/ en/ stable/ development/ window s-provisioning/](https://osquery.readthedocs.io/en/stable/development/window-s-provisioning/)

[https:// osquery.readthedocs.io/ en / stable/ installation/ custom- packages/](https://osquery.readthedocs.io/en/stable/installation/custom-packages/)

Performance safety

Tip

[http:// osquery.io/](http://osquery.io/)

[https:// github.com/ face
book/ osquery](https://github.com/facebook/osquery)


[https:// osquery.readthe
docs.io/ en/ stable/](https://osquery.readthedocs.io/en/stable/)

```
$ sudo -E python ./tools/analysis/profile.py --config osquery.conf
Profiling query: SELECT * FROM kernel_extensions WHERE name NOT LIKE 'com.apple.%' AND name != '__kernel__';
D:0 C:0 M:0 F:0 U:1 non_apple_kexts (1/1): duration: 0.519426107407 cpu_time: 0.096729864 memory: 6447104 fds: 5
Profiling query: SELECT name, path, bundle_version, minimum_system_version, applescript_enabled, bundle_executable FROM
D:0 C:0 M:0 F:0 U:1 installed_applications (1/1): duration: 0.507317066193 cpu_time: 0.113432314 memory: 7639040
Profiling query: SELECT service, process FROM alf_services WHERE state != 0;
D:0 C:0 M:0 F:0 U:0 alf_services (1/1): duration: 0.525090932846 cpu_time: 0.021108868 memory: 5406720 fds: 5 ut
Profiling query: SELECT * FROM processes WHERE on_disk != 1;
D:0 C:0 M:0 F:0 U:0 processes_not_on_disk (1/1): duration: 0.521270990372 cpu_time: 0.030440911 memory: 6148096
Profiling query: SELECT name, version FROM kernel_extensions;
D:0 C:0 M:0 F:0 U:1 all_kexts (1/1): duration: 0.522475004196 cpu_time: 0.089579066 memory: 6500352 fds: 5 utili
Profiling query: SELECT DISTINCT process.name, listening.port, listening.protocol, listening.family, listening.address
D:2 C:1 M:0 F:0 U:2 processes_binding_to_ports (1/1): duration: 1.02116107941 cpu_time: 0.668809664 memory: 6340
```

PACKS: do not re-invent the wheel!


Branch: master ▾ osquery / packs /

Create new fileUpload filesFind fileHistory

 Chaz6 and obelisk Fix typos in packs/windows-hardening.conf (#4282)


Latest commit 67dd49a 2 days ago

..

 hardware-monitoring.conf


Remove duplicate mode column in device_nodes query (#4107)

2 months ago

 incident-response.conf


packs: adding platform tag incident-response pack (#4155)

2 months ago

 it-compliance.conf


Updated to scope all users by default (#3736)

7 months ago

 osquery-monitoring.conf


config: Allow scheduled queries to set blacklist=false (#4005)

4 months ago

 ossec-rootkit.conf


Querypack equivalent of ossec rootkit db (#3377)

10 months ago

 osx-attacks.conf


Update osx-attacks.conf (#4218)

6 days ago

 unwanted-chrome-extensions.conf


packs: Adding a pack for unwanted chrome extensions (#3889)

6 months ago

 vuln-management.conf


packs: fixing backdoored python pack (#3707)

7 months ago

 windows-attacks.conf

packs: remove escape - Error parsing the "windows-attacks" pack JSON (#...

a month ago

 windows-hardening.conf

Fix typos in packs/windows-hardening.conf (#4282)

2 days ago

Deploy strategy

```
1 {
2   "options": {
3     "schedule_splay_percent": 10
4   },
5   "platform": "darwin",
6   "schedule": {
7     "crontab": {
8       "query": "SELECT * FROM crontab",
9       "interval": 86400,
10      "description": "Show crontab entries for root user"
11    },
12    "os_version": {
13      "query": "SELECT * FROM os_version;",
14      "interval": 86400,
15      "description": "Record the current version of the OS",
16      "snapshot": true
17    },
18    "hardware_events": {
19      "query": "SELECT * FROM hardware_events;",
20      "interval": 3600,
21      "description": "Hardware (PCI/USB/HID) events",
22      "removed": false
23    }
24  },
25  "file_paths": {
26    "configuration": [
27      "/etc/%%"
28    ],
29    "binaries": [
30      "/usr/bin/%%",
31      "/usr/sbin/%%",
32      "/bin/%%",
33      "/sbin/%%",
34      "/usr/local/bin/%%",
35      "/usr/local/sbin/%%",
36      "/opt/bin/%%",
37      "/opt/sbin/%%"
38    ]
39  }
40 }
```

Splay the query interval by 10%

Destination platform(s). Can have multiple. Ex: "darwin, windows"

Query interval. Note: interval timer only advances while a host is online. Consider shorter intervals for hosts that aren't always active.

Enable snapshot mode for this query

Differential mode, but ignores "removed" events

Define file paths to be monitored by the file_events table

Log collection

Osquery is designed to work with any environment's existing data infrastructure. Since the problem space of forwarding logs is so well developed, osquery does not implement log forwarding internally.

Kibana , splunk, Rsyslog, Fluentd, Scribe, etc.



Tip

[https:// osquery
.readthedocs.io/
en/ stable/ depl
oyment/ log-
aggregation/](https://osquery.readthedocs.io/en/stable/deployment/log-aggregation/)

Live queries

<https://github.com/mwielgoszewski/doorma>

doorman nodes packs queries **distributed** files tags rules add ▾

distributed queries

| node | query | created | run after | retrieved | status | result |
|--------------------------------------|--|----------------------------|-------------|----------------------------|----------|----------------------------|
| ECF5B6C8-67CB-4333-A805-B6DD47297A90 | select encrypted from disk_encryption join mounts on disk_encryption.name = mounts.device where mounts.path = '/' AND encrypted = 0; | 2016-05-18 01:44:51.018149 | immediately | | NEW | |
| 033D02C8-C8FE-451F-BE03-DDDF00B10260 | select encrypted from disk_encryption join mounts on disk_encryption.name = mounts.device where mounts.path = '/' AND encrypted = 0; | 2016-05-18 01:44:51.018149 | immediately | 2016-05-18 01:45:05.177017 | COMPLETE | — |
| F5720272-528B-4AAS-A206-D9D311FB4579 | select encrypted from disk_encryption join mounts on disk_encryption.name = mounts.device where mounts.path = '/' AND encrypted = 0; | 2016-05-18 01:44:51.018149 | immediately | 2016-05-18 01:45:05.178201 | PENDING | |
| 17AAB828-42DB-596F-A900-79CC41FFD68F | select * from osquery_info; | 2016-05-17 15:36:30.415320 | immediately | 2016-05-17 15:41:00.614074 | COMPLETE | 2016-05-17 15:41:00.663748 |
| 17AAB828-42DB-596F-A900-79CC41FFD68F | select * from browser_plugins; | 2016-05-17 15:10:34.255687 | immediately | 2016-05-17 15:10:52.010617 | COMPLETE | 2016-05-17 15:10:52.069765 |
| 17AAB828-42DB-596F-A900-79CC41FFD68F | select * from usb_devices; | 2016-05-17 14:18:13.684462 | immediately | 2016-05-17 14:19:00.163770 | COMPLETE | 2016-05-17 14:19:00.191692 |
| 17AAB828-42DB-596F-A900-79CC41FFD68F | select * from usb_devices where removable = 1; | 2016-05-17 14:17:46.551588 | immediately | 2016-05-17 14:17:47.391725 | COMPLETE | 2016-05-17 14:17:47.416616 |
| 17AAB828-42DB-596F-A900-79CC41FFD68F | select * from osquery_info; | 2016-05-17 14:17:27.440479 | immediately | 2016-05-17 14:17:27.662393 | COMPLETE | 2016-05-17 14:17:27.711335 |

displaying 1 - 12 of 12 distributed queries

doorman **nodes** packs queries distributed files tags rules add ▾

active nodes / inactive nodes

| Host Identifier | Node Key | Name | Make | Model | Serial | Cpu Cores | Memory | Last IP Address | Enrolled Date | Last Check-in Date | Tags | |
|-----------------|--------------------------------------|--------------|------------|----------------|--------------|---|--------|-----------------|-----------------|----------------------------|----------------------------|---|
| unpleated | bb795b79-1d12-4eda-8778-cbb956800c2f | unpleated | Apple Inc. | MacBookPro11,3 | FF3XR75J709 | Intel(R) Core(TM) i7-4980HQ CPU @ 2.80GHz | 4 | 17179869184 | 117.65.103.49 | 2016-05-18 01:43:21.378974 | 2016-07-01 19:27:12.528095 | servers x web x |
| | | | | | | | | | | | | |
| collectasia | 15577f94-cdf2-488a-b213-c1603551b658 | collectasia | Apple Inc. | MacBookPro11,3 | HFO8X1E0XG99 | Intel(R) Core(TM) i7-4980HQ CPU @ 2.80GHz | 4 | 17179869184 | 86.106.186.130 | 2016-05-18 01:43:21.872970 | 2016-07-01 19:27:12.537671 | desktop x support x |
| | | | | | | | | | | | | |
| costotome | 1cd9dc6f-3bbe-458a-8861-d8b99ed7e546 | costotome | Apple Inc. | MacBookPro11,3 | M973QG4P2MS1 | Intel(R) Core(TM) i7-4980HQ CPU @ 2.80GHz | 4 | 17179869184 | 156.173.174.183 | 2016-05-18 01:43:22.244124 | 2016-06-25 05:49:03.118553 | laptops x exe x |
| | | | | | | | | | | | | |
| strickenness | 7065b3bb-092f-4a35-9660-61dd8d721338 | strickenness | Apple Inc. | MacBookPro11,3 | 13Xh2EO645J4 | Intel(R) Core(TM) i7-4980HQ CPU @ 2.80GHz | 4 | 17179869184 | 123.104.11.170 | 2016-05-18 01:43:22.588424 | 2016-07-01 19:27:12.531354 | laptops x |
| | | | | | | | | | | | | |
| subzonal | 16aaf6c6-deb4-41b3-a68d-273a06a3af29 | subzonal | Apple Inc. | MacBookPro11,3 | USO8TXG8XQ14 | Intel(R) Core(TM) i7-4980HQ CPU @ 2.80GHz | 4 | 17179869184 | 80.253.12.74 | 2016-05-18 01:43:22.869550 | 2016-07-01 19:27:12.534523 | servers x |
| | | | | | | | | | | | | |
| | | anic | Apple Inc. | MacBookPro11,3 | 8h7HKWXLTOH | Intel(R) Core(TM) i7-4980HQ CPU @ 2.80GHz | 4 | 17179869184 | 178.192.91.225 | 2016-06-13 14:01:10.957212 | 2016-07-01 19:27:12.544246 | |
| | | romy | Apple Inc. | MacBookPro11,3 | JA5911D991H | Intel(R) Core(TM) i7-4980HQ CPU @ 2.80GHz | 4 | 17179869184 | 117.132.20.72 | 2016-06-15 03:27:51.386064 | 2016-07-01 19:27:12.547905 | |
| | status | result | | | | | | | | | | |

displaying 1 - 7 of 7 active nodes 📌

What about global log collection?

Firehose on AWS

```
{
  "options": {
    "host_identifier": "hostname",
    "schedule_splay_percent": 10,
    "logger_plugin": "aws_kinesis,aws_firehose",
    "aws_kinesis_stream": "foo_stream",
    "aws_firehose_stream": "bar_delivery_stream",
    "aws_access_key_id": "ACCESS_KEY",
    "aws_secret_access_key": "SECRET_KEY",
    "aws_region": "us-east-1"
  },
  "schedule": {
    "time": {
      "query": "SELECT * FROM time;",
      "interval": 2,
      "removed": false
    }
  }
}
```

Inference and baseline: what we can realise
if you take the corner cases?



Tip

[http:// osquery.i
o/](http://osquery.io/)

[https:// github.com/ face
book/ osquery](https://github.com/facebook/osquery)

[https:// osquery.readthed
ocs.io/ en/ stable/](https://osquery.readthedocs.io/en/stable/)

Update and plugin

- * How can I update the application?
- * What about my unique logs file and event ?

Alternatives?

Facter : Collect and display system facts <https://tickets.puppet.com/browse/FACT>

Sysdig: Open Source Troubleshooting, Forensics, and Security

<https://sysdig.com/opensource/>



Questions?

- Fabio Nigi - nigifabio@gmail.com
- <http://osquery.io/>
- <https://github.com/facebook/osquery>
- <https://osquery-slack.herokuapp.com>