osquery: Cross-platform Lightweight Performant Host Visibility

Teddy Reed
Facebook
@teddyreedv

Sereyvathana Ty
Facebook
@sereyvathanaty

osquery is an agent

```
$ ps ax | grep osquery
15658  /usr/local/bin/osqueryd
```

Why build osquery?

We need more than process auditing on OS X (and others)

We need an additional signal for fleet inventory truth

Small team of security engineers looking for a one-size-fits-all

Our *customers* are performance aware software developers

What machines have the chrome extension **xyz123** installed?

What are the **top** 50 most unique kernel modules?

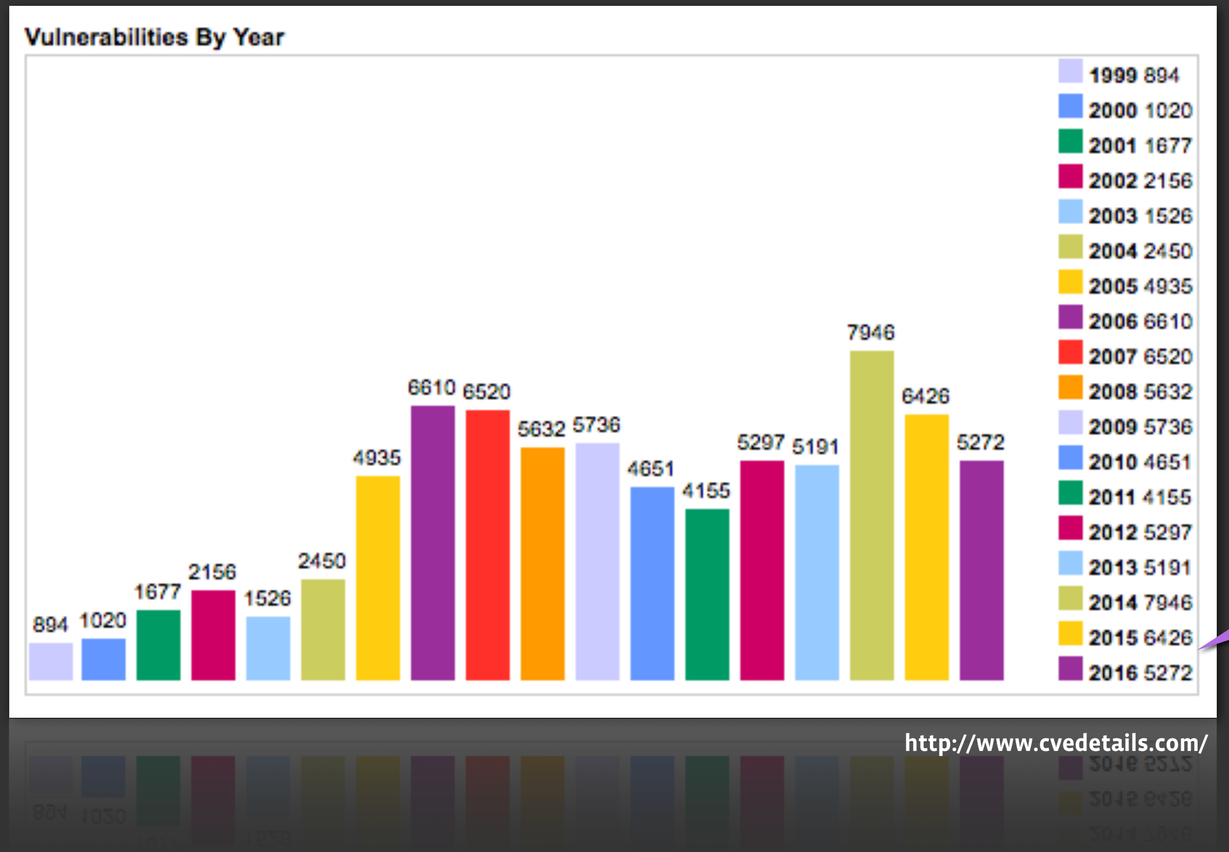
Did anyone download a file named **abc789** yesterday?

Is anything bridging routes from VPN to their LAN?

Can I graph the number of mounts or open
file descriptors every machine had yesterday by hour?

Aside: CVEs Everywhere

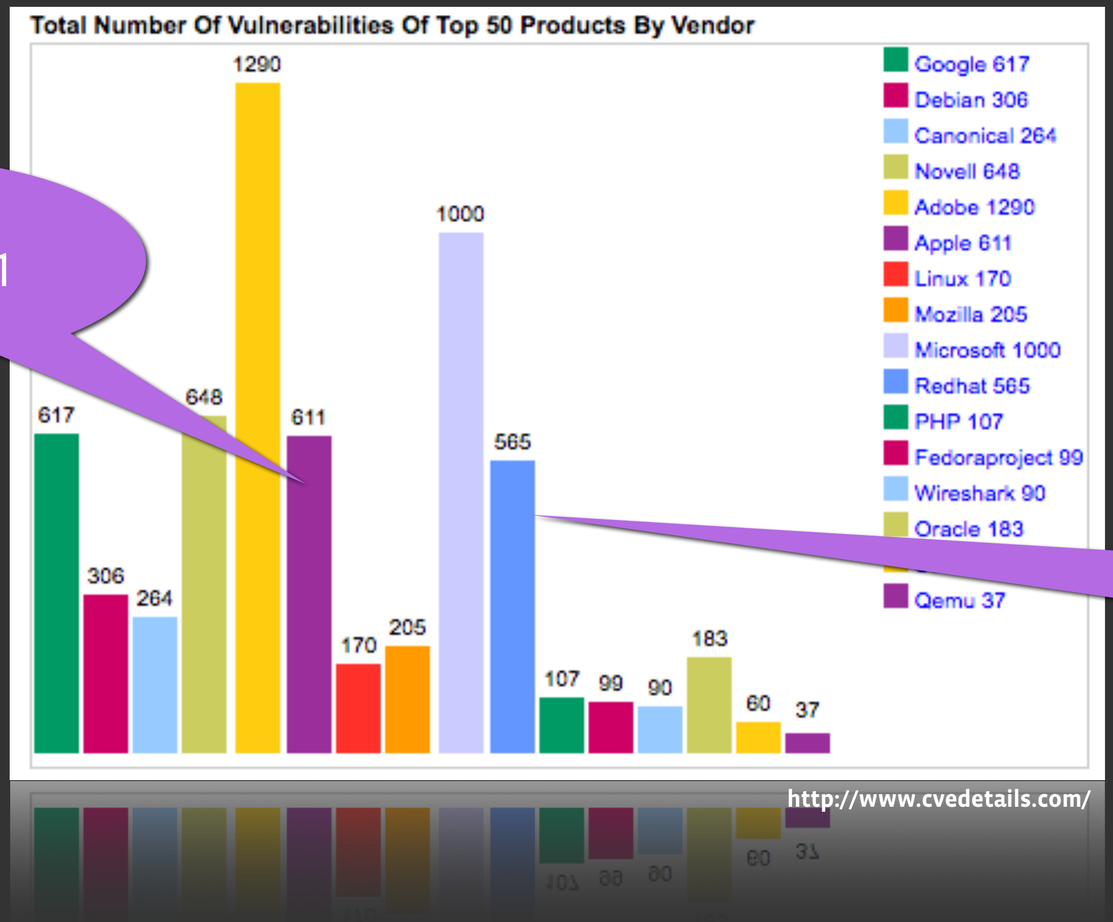
Running third-party applications and libraries introduces risk in our enterprise (and in production)!



5-7K reported

Aside: CVEs Everywhere

Running third-party applications and libraries
introduces risk in our enterprise (and in production)!



Apple, 611

Redhat, 565

Response after vulnerability announcements

What machines are running the affected versions?

Estimate on deployability from fleet management team?

How long until affected hosts have updated, meaning vulnerability mitigated?

Requirements: you have a client inventory, you have the new version to test, you can deploy software

Ground truth for client inventory

In your enterprise, group by each OS and OS patch?

For each OS, how many applications are installed?

Have you labeled your hosts somewhere like DHCP or 802.1x?

Asset management is complex and can be decorated with network and host events. Augment or begin your inventory using host based sensors.

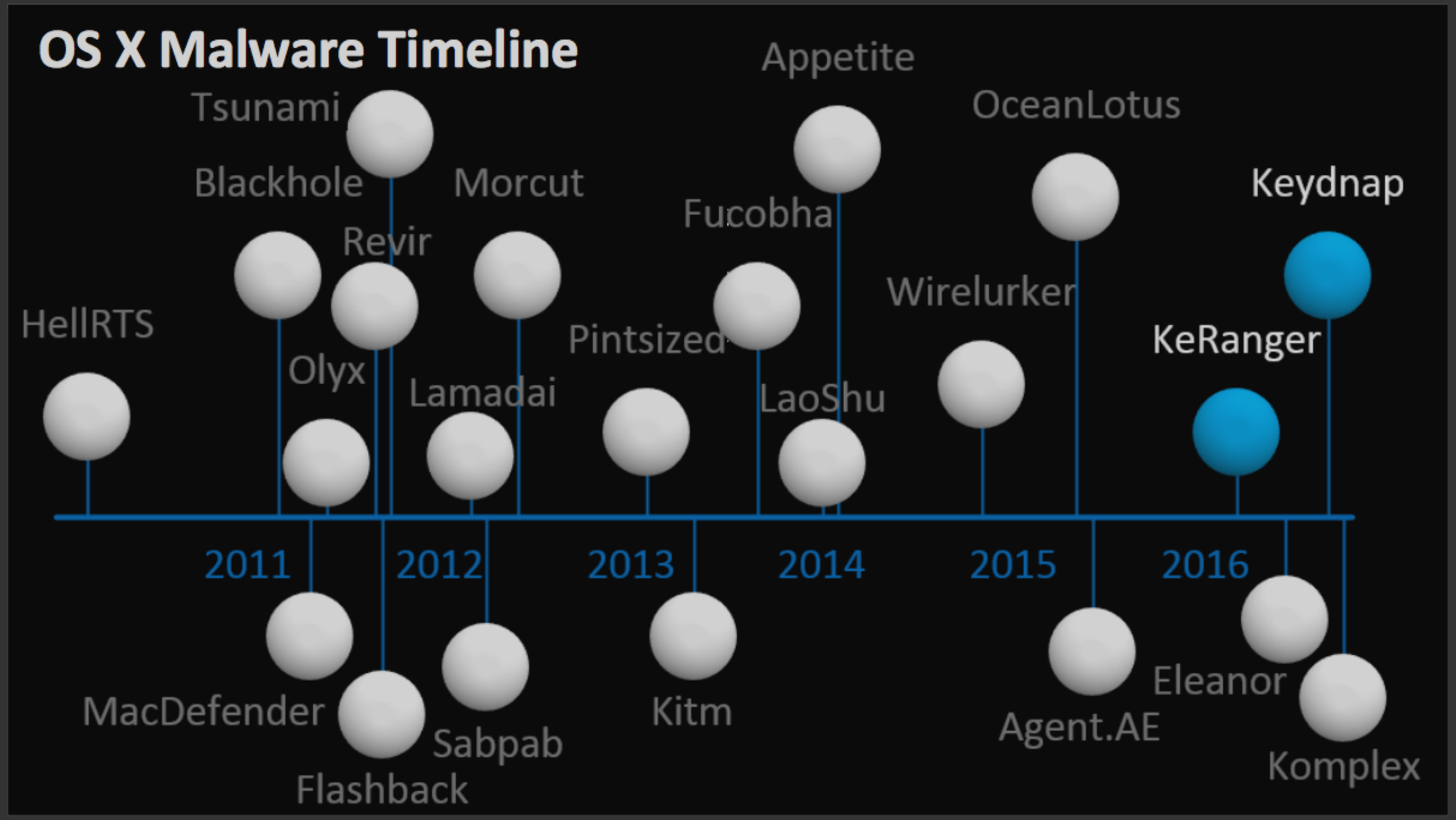
Malware detection using simple IOCs

Simple malware families on OS X have simple patterns

YARA, hashing, fuzzy hashing is not needed

Invariant detection on OS X through metadata is easy

Non-standard, yet obvious, types of IOCs are abundant on OS X. Simple and easy attacks are social and manipulate logic vulnerabilities, they have few indicators of exploitation initially



https://www.virusbulletin.com/uploads/pdf/conference_slides/2016/Kalnai_Jirkal-vb-2016-malicious-osx-cocktail.pdf

KeRanger

Started process named `kernel_service`
Drop file `/Users/*/Library/.kernel_*`

Keydnap

Created launched service called
`com.apple.iCloud.sync.daemon`

Now let’s introduce **osquery** to solve goals

What is osquery?

Explore your operating system using SQL

Host visibility motivated by intrusion detection

100% OS API usage, no fork execve

Facebook's host intrusion detection agent

Why use SQL?

OS concepts are shared on Mac, Linux, and Windows

the “concepts” have attributes:
user ids, process ids, descriptors, ports, paths

most developers and administrators know SQL

Why use SQL?

[concept]

```
SELECT pid, name, uid FROM processes
```

Why use SQL?

[attributes] [concept]

```
SELECT pid, name, uid FROM processes
```

Why use SQL?

[attributes]

```
SELECT pid, name, uid FROM processes
```

```
WHERE uid != 0
```

[constraints]

Why use SQL?

[attribute]

```
SELECT pid, name, username FROM processes
JOIN users ON processes.uid = users.uid
WHERE uid != 0
```

[join]

Over 100 tables to join

(https://osquery.io/docs/tables/)

- acpi_tables
- arp_cache
- crontab
- file_events
- kernel_info
- listening_ports
- logged_in_users
- mounts
- pci_devices
- processes
- routes
- shell_history
- smbios_tables
- suid_bin
- system_controls
- usb_devices
- users
- groups
- rpm_packages
- apt_sources
- deb_packages
- homebrew_packages
- kernel_modules
- memory_map
- shared_memory
- browser_plugins
- startup_items

arp_cache

A constant view should optimize data collection

```
$ osqueryi “select * from arp_cache”
+-----------------+-------------------+-----------+-----------+
| address         | mac               | interface | permanent |
+-----------------+-------------------+-----------+-----------+
| 172.16.103.255  | incomplete        | vmnet1    | 0         |
| 172.24.40.1     | 00:00:0c:9f:f1:2c | en0       | 0         |
| 172.24.47.255   | incomplete        | en0       | 0         |
| 192.168.146.181 | 00:0c:29:e4:38:42 | vmnet8    | 0         |
| 192.168.146.255 | incomplete        | vmnet8    | 0         |
| 224.0.0.251     | 01:00:5e:00:00:fb | en0       | 1         |
| 239.255.255.250 | 01:00:5e:7f:ff:fa | en0       | 1         |
| 255.255.255.255 | incomplete        | en0       | 0         |
+-----------------+-------------------+-----------+-----------+
```

```
$ ping 172.24.40.38 -c 1 >/dev/null
$ osqueryi “select * from arp_cache”
+-----------------+-------------------+-----------+-----------+
| address         | mac               | interface | permanent |
+-----------------+-------------------+-----------+-----------+
| 172.16.103.255  | incomplete        | vmnet1    | 0         |
| 172.24.40.1     | 00:00:0c:9f:f1:2c | en0       | 0         |
| 172.24.40.38    | 6c:40:08:99:8f:38 | en0       | 0         |
| 172.24.47.255   | incomplete        | en0       | 0         |
| 192.168.146.181 | incomplete        | vmnet8    | 0         |
| 192.168.146.255 | incomplete        | vmnet8    | 0         |
| 224.0.0.251     | 01:00:5e:00:00:fb | en0       | 1         |
| 239.255.255.250 | 01:00:5e:7f:ff:fa | en0       | 1         |
| 255.255.255.255 | incomplete        | en0       | 0         |
+-----------------+-------------------+-----------+-----------+
```

arp_cache

A constant view should optimize data collection

```
$ osqueryi “select * from arp_cache”
+-----------------+-------------------+-----------+-----------+
| address         | mac               | interface | permanent |
+-----------------+-------------------+-----------+-----------+
| 172.16.103.255  | incomplete        | vmnet1    | 0         |
| 172.24.40.1     | 00:00:0c:9f:f1:2c | en0       | 0         |
| 172.24.47.255   | incomplete        | en0       | 0         |
| 192.168.146.181 | 00:0c:29:e4:38:42 | vmnet8    | 0         |
| 192.168.146.255 | incomplete        | vmnet8    | 0         |
| 224.0.0.251     | 01:00:5e:00:00:fb | en0       | 1         |
| 239.255.255.250 | 01:00:5e:7f:ff:fa | en0       | 1         |
| 255.255.255.255 | incomplete        | en0       | 0         |
+-----------------+-------------------+-----------+-----------+
```

arp_cache

A constant view should optimize data collection

```
$ ping 172.24.40.38 -c 1 >/dev/null
$ osqueryi “select * from arp_cache”
+-----------------+-------------------+-----------+-----------+
| address         | mac               | interface | permanent |
+-----------------+-------------------+-----------+-----------+
| 172.16.103.255  | incomplete        | vmnet1    | 0         |
| 172.24.40.1     | 00:00:0c:9f:f1:2c | en0       | 0         |
| 172.24.40.38    | 6c:40:08:99:8f:38 | en0       | 0         |
| 172.24.47.255   | incomplete        | en0       | 0         |
| 192.168.146.181 | incomplete        | vmnet8    | 0         |
| 192.168.146.255 | incomplete        | vmnet8    | 0         |
| 224.0.0.251     | 01:00:5e:00:00:fb | en0       | 1         |
| 239.255.255.250 | 01:00:5e:7f:ff:fa | en0       | 1         |
| 255.255.255.255 | incomplete        | en0       | 0         |
+-----------------+-------------------+-----------+-----------+
```

arp_cache

Apply set-difference to ‘most’ SELECTs

```
$ cat /var/log/osquery/osqueryd.results.log
{
  “name”: ”…”,
  “hostIdentifier”: ”…”,
  “calendarTime”: "...",
  “unixTime”: "1475002120",
  “columns”:{
    “address”: "172.24.40.38",
    “mac”: "6c:40:08:99:8f:38",
    “interface”: "en0",
    “permanent”: ”0”
  },
  “action”: "added"
}
```

Turn snapshot-in-time views into event streams

Execute SELECT queries in a schedule

```
{
 "options": {
   "disable_audit": "false",
   "audit_allow_config": "true"
  },

 "schedule": {
    "arp_cache_changes": {
      "query": "select * from arp_cache",
      "interval": 60,
      "removed": false
    }
  }
}
```

Fun query time!

```
osquery> SELECT * FROM listening_ports JOIN processes USING (pid);
```

```
osquery> SELECT * FROM listening_ports JOIN processes USING (pid);
+------+-------+----------+-----------+-----------------+
| pid  | port  | protocol | address   | name            |
+------+-------+----------+-----------+-----------------+
| 5139 | 17500 | 6        | 0.0.0.0   | dropbox         |
| 5183 | 17501 | 6        | 0.0.0.0   | dropbox         |
| 5139 | 17600 | 6        | 127.0.0.1 | dropbox         |
| 5183 | 17501 | 6        | ::        | dropbox         |
| 5183 | 17500 | 17       | 0.0.0.0   | dropbox         |
| 6181 | 5353  | 17       | 0.0.0.0   | chrome          |
| 6181 | 5353  | 17       | ::        | chrome          |
| 4774 | 0     | 0        |           | gnome-session-b |
+------+-------+----------+-----------+-----------------+
```

```
osquery> SELECT * FROM users JOIN chrome_extensions USING (uid);
```

```
osquery> SELECT * FROM users JOIN chrome_extensions USING (uid);
+----------+--------------------------------+----------------+
| username | identifier                     | version        |
+----------+--------------------------------+----------------+
| reed     | aapocclcgogkmnckokdopfmhonfmgoek | 0.9          |
| reed     | bhmmomiinigofkjcapegjjndpbikblnp | 3.0.6        |
| reed     | blpcfgokakmgnkcojhhkbfbldkacnbeo | 4.2.8        |
| reed     | gbchcmhmhahfdphkhkmpfmihenigjmpp | 52.0.2743.48 |
| reed     | laookkfknpbbblfpciffpaejjkokdgca | 0.91.5       |
| reed     | nmmhkkegccagdldgiimedpiccmgmieda | 1.0.0.0      |
| reed     | pjkljhegncpnkpknbcohdijeoejaedia | 8.1          |
| reed     | pkedcjkdefgpdelpbcmbmeomcjbeemfm | 5316.725.0.15 |
+----------+--------------------------------+----------------+
```

```
osquery> SELECT filename, size, mode, sha256
    ...> FROM file
    ...> JOIN hash USING (path) WHERE file.directory = '/boot';
```

```
osquery> SELECT filename, size, mode, sha256
    ...> FROM file
    ...> JOIN hash USING (path) WHERE file.directory = '/boot';
+-----------------------------------+----------+------+--------+
| filename                          | size     | mode | sha256 |
+-----------------------------------+----------+------+--------+
| System.map-4.4.0-45-generic       | 3869895  | 0600 | ...    |
| vmlinuz-4.4.0-45-generic          | 7054208  | 0600 | ...    |
| memtest86+.elf                    | 184380   | 0644 | ...    |
| vmlinuz-4.4.0-45-generic.efi.signed | 7056120  | 0600 | ...    |
| initrd.img-4.4.0-45-generic       | 40005488 | 0644 | ...    |
| config-4.4.0-45-generic           | 189760   | 0644 | ...    |
+-----------------------------------+----------+------+--------+
```

```
osquery> SELECT * FROM arp_cache;
+-------------+-------------------+-----------+-----------+
| address     | mac               | interface | permanent |
+-------------+-------------------+-----------+-----------+
| 172.20.10.1 | 3a:ha:ha:98:43:64 | wlp4s0    | 0         |
+-------------+-------------------+-----------+-----------+

osquery> SELECT * FROM (
    ...>   SELECT COUNT(1) AS mac_count, mac
    ...>   FROM arp_cache
    ...>   GROUP BY mac
    ...> ) WHERE mac_count > 1;
```

```
osquery> SELECT * FROM arp_cache;
+-------------+-------------------+-----------+-----------+
| address     | mac               | interface | permanent |
+-------------+-------------------+-----------+-----------+
| 172.20.10.1 | 3a:ha:ha:98:43:64 | wlp4s0    | 0         |
+-------------+-------------------+-----------+-----------+

osquery> SELECT * FROM (
    ...>   SELECT COUNT(1) AS mac_count, mac
    ...>   FROM arp_cache
    ...>   GROUP BY mac
    ...> ) WHERE mac_count > 1;
```

arp spoofing!!!

```
osquery> SELECT codename, version, major, minor FROM os_version;
+----------+-----------------------------+-------+-------+
| codename | version                     | major | minor |
+----------+-----------------------------+-------+-------+
| xenial   | 16.04.1 LTS (Xenial Xerus) | 16    | 4     |
+----------+-----------------------------+-------+-------+
osquery> SELECT hostname, physical_memory, cpu_logical_cores,
    ...> hardware_model AS model FROM system_info;
+----------+-----------------+--------------------+------------+
| hostname | physical_memory | cpu_logical_cores  | model      |
+----------+-----------------+--------------------+------------+
| laptop2  | 20783439872     | 4                  | 20FAS22W00 |
+----------+-----------------+--------------------+------------+
```

```
osquery> .mode line
osquery> .all platform_info;
     vendor = Apple Inc.
    version = MBP121.88Z.0167.B17.1606231721
       date = 06/23/2016
   revision =
    address = 0xff990000
       size = 8388608
volume_size = 1507328
      extra =
osquery>
```

```
osquery> .timer on
osquery> SELECT COUNT(1) FROM apps;
count(1) = 411
Run Time: real 0.244 user 0.163018 sys 0.071432
osquery> SELECT COUNT(1) FROM processes;
count(1) = 429
Run Time: real 0.054 user 0.011647 sys 0.028173
osquery> SELECT COUNT(1) FROM launchd;
count(1) = 578
Run Time: real 0.239 user 0.106277 sys 0.086952
```

Crazy query time!

Authorities using non-RSA signature algorithms

```
SELECT
    common_name,
    self_signed,
    key_strength,
    key_algorithm,
    signing_algorithm
FROM certificates
WHERE
    ca = 1 AND
    signing_algorithm NOT LIKE '%WithRSAEncryption';
```

```
osquery> select common_name, self_signed, key_strength, key_algorithm, signing_algorithm \
   ...> from certificates where ca = 1 and signing_algorithm NOT like '%WithRSAEncryption';
+-----------------------------------------------------------------+-------------+--------------+----------------+--------------------+
| common_name                                                     | self_signed | key_strength | key_algorithm  | signing_algorithm  |
+-----------------------------------------------------------------+-------------+--------------+----------------+--------------------+
| AffirmTrust Premium ECC                                         | 1           | secp384r1    | id-ecPublicKey | ecdsa-with-SHA384  |
| Apple Root CA - G3                                              | 1           | secp384r1    | id-ecPublicKey | ecdsa-with-SHA384  |
| DigiCert Assured ID Root G3                                     | 1           | secp384r1    | id-ecPublicKey | ecdsa-with-SHA384  |
| DigiCert Global Root G3                                         | 1           | secp384r1    | id-ecPublicKey | ecdsa-with-SHA384  |
| Entrust Root Certification Authority - EC1                      | 1           | secp384r1    | id-ecPublicKey | ecdsa-with-SHA384  |
| GeoTrust Primary Certification Authority - G2                   | 1           | secp384r1    | id-ecPublicKey | ecdsa-with-SHA384  |
| GlobalSign                                                      | 1           | prime256v1   | id-ecPublicKey | ecdsa-with-SHA256  |
| GlobalSign                                                      | 1           | secp384r1    | id-ecPublicKey | ecdsa-with-SHA384  |
| Symantec Class 1 Public Primary Certification Authority - G4    | 1           | secp384r1    | id-ecPublicKey | ecdsa-with-SHA384  |
| Symantec Class 2 Public Primary Certification Authority - G4    | 1           | secp384r1    | id-ecPublicKey | ecdsa-with-SHA384  |
| Symantec Class 3 Public Primary Certification Authority - G4    | 1           | secp384r1    | id-ecPublicKey | ecdsa-with-SHA384  |
| VeriSign Class 3 Public Primary Certification Authority - G4    | 1           | secp384r1    | id-ecPublicKey | ecdsa-with-SHA384  |
| thawte Primary Root CA - G2                                     | 1           | secp384r1    | id-ecPublicKey | ecdsa-with-SHA384  |
+-----------------------------------------------------------------+-------------+--------------+----------------+--------------------+
```

Default routes to non-enterprise gateways

```
WITH routes_to_corporate_network AS (
    SELECT * FROM routes WHERE interface IN (
    SELECT interface FROM (
     SELECT
       interface,
       address,
       inet_aton(address) AS n,
       inet_aton('172.16.0.0') min,
       inet_aton('172.31.255.255') as max
     FROM interface_addresses
     WHERE n > min AND n < max)
   ) AND type = 'static' AND gateway <> '127.0.0.1')
SELECT destination rd, gateway rg FROM routes
WHERE destination = '0.0.0.0' AND gateway NOT IN (
  SELECT gateway FROM routes_to_corporate_network);
```

Default routes to non-enterprise gateways

```
WITH routes_to_corporate_network AS (
    SELECT * FROM routes WHERE interface IN (
    SELECT interface FROM (
     SELECT
       interface,
       address,
       inet_aton(address) AS n,
       inet_aton('172.16.0.0') min,
       inet_aton('172.31.255.255') as max
     FROM interface_addresses
     WHERE n > min AND n < max)
   ) AND type = 'static' AND gateway <> '127.0.0.1')
SELECT destination rd, gateway rg FROM routes
WHERE destination = '0.0.0.0' AND gateway NOT IN (
  SELECT gateway FROM routes_to_corporate_network);
```

Default routes to non-enterprise gateways

```
WITH routes_to_corporate_network AS (
    SELECT * FROM routes WHERE interface IN (
    SELECT interface FROM (
     SELECT
       interface,
       address,
       inet_aton(address) AS n,
       inet_aton('172.16.0.0') min,
       inet_aton('172.31.255.255') as max
     FROM interface_addresses
     WHERE n > min AND n < max)
   ) AND type = 'static' AND gateway <> '127.0.0.1')
SELECT destination rd, gateway rg FROM routes
WHERE destination = '0.0.0.0' AND gateway NOT IN (
  SELECT gateway FROM routes_to_corporate_network);
```

Default routes to non-enterprise gateways

```
WITH routes_to_corporate_network AS (
    SELECT * FROM routes WHERE interface IN (
    SELECT interface FROM (
     SELECT
       interface,
       address,
       inet_aton(address) AS n,
       inet_aton('172.16.0.0') min,
       inet_aton('172.31.255.255') as max
     FROM interface_addresses
     WHERE n > min AND n < max)
   ) AND type = 'static' AND gateway <> '127.0.0.1')
SELECT destination rd, gateway rg FROM routes
WHERE destination = '0.0.0.0' AND gateway NOT IN (
  SELECT gateway FROM routes_to_corporate_network);
```

```
osquery> WITH routes_to_corporate_network AS (
    ...>    SELECT * FROM routes WHERE interface IN (
    ...>    SELECT interface FROM (
    ...>      SELECT
    ...>        interface,
    ...>        address,
    ...>        inet_aton(address) AS n,
    ...>        inet_aton('172.16.0.0') min,
    ...>        inet_aton('172.31.255.255') as max
    ...>      FROM interface_addresses
    ...>      WHERE n > min AND n < max)
    ...>    ) AND type = 'static' AND gateway <> '127.0.0.1')
    ...> SELECT destination rd, gateway rg FROM routes
    ...> WHERE destination = '0.0.0.0' AND gateway NOT IN (
    ...>   select gateway FROM routes_to_corporate_network);

osquery>
```

```
osquery> WITH RECURSIVE
    ...>   xaxis(x) AS (VALUES(-2.0) UNION ALL SELECT x+0.05 FROM xaxis WHERE x<1.2),
    ...>   yaxis(y) AS (VALUES(-1.0) UNION ALL SELECT y+0.1 FROM yaxis WHERE y<1.0),
    ...>   m(iter, cx, cy, x, y) AS (SELECT 0, x, y, 0.0, 0.0 FROM xaxis, yaxis UNION ALL
    ...>     SELECT iter+1, cx, cy, x*x-y*y + cx, 2.0*x*y + cy FROM m WHERE (x*x + y*y) < 4.0 AND iter<28),
    ...>   m2(iter, cx, cy) AS (SELECT max(iter), cx, cy FROM m GROUP BY cx, cy),
    ...>   a(t) AS (SELECT group_concat( substr(' .+*#', 1+min(iter/7,4), 1), '') FROM m2 GROUP BY cy)
```

```
osquery> SELECT group_concat(rtrim(t),x'0a') FROM a;
  group_concat(rtrim(t),x'0a') =
                                      ....#
                                     ..#*..
                                   ..+####+.
                                .......+####....   +
                               ..##+*##########+.++++
                              .+.##################+.
                    ............+####################+.+
                    ..++..#.....*######################+.
                   ...+########++#########################.
                ....+*###################################.
 ##########################################################...
                ....+*###################################.
                   ...+########++#########################.
                    ..++..#.....*######################+.
                    ............+####################+.+
                              .+.##################+.
                               ..##+*##########+.++++
                                .......+####....   +
                                   ..+####+.
                                     ..#*..
                                      ....#
                                      +.
```

File integrity monitoring

File integrity monitoring: events tables

/etc/osquery/osquery.conf

```
"file_paths": {
    "homes": ["/home/*"]
}
```

Some tables end with `_events`

These capture data in real time and report it during query-time

File integrity monitoring: events tables

/etc/osquery/osqu

```
"file_paths": {
    "homes": ["/home/**"]
}
```

What changed?

Some tables end with `_events`

These capture data in real time and report it during query-time

File integrity monitoring: events tables

/etc/osquery/osquery.conf

```
"file_paths": {
    "homes": ["/home/**"],
    "etc": [
       "/etc/ssh/*",
       "/etc/mach_init.d/*",
       "/etc/security/*",
       "/etc/*"
    ]
}
```

Define sets of path globbing expressions

Use the named-sets throughout osquery, such as for YARA scanning

File integrity monitoring: events tables

```
$ osqueryi --verbose --nodisable_events
[...] Added file event listener to: /private/etc/ssh/*
[...] Added file event listener to: /private/etc/mach_init.d/*
[...] Added file event listener to: /private/etc/security/*
[...] Added file event listener to: /private/etc/*
[...] Added file event listener to: /home/**
osquery> select * from file_events;
osquery> CTRL+Z
[1]  + 60644 suspended  osqueryi --verbose --nodisable_events
[146] $
$ touch ~/hello-osdf2016
$ fg
[1]  + 60644 continued  osqueryi --verbose --nodisable_events
osquery> select * from file_events;
```

Demo: hardware events!

Plugins, extensions, modules, and more

```
namespace osquery {
namespace tables {

QueryData genTime(QueryContext& ctx) {
  QueryData results;
  struct tm* now = localtime(time(0));

  Row r;
  r["hour"] = INTEGER(now->tm_hour);
  r["minutes"] = INTEGER(now->tm_min);
  r["seconds"] = INTEGER(now->tm_sec);
  results.push_back(r);

  return results;
}
}
}
```

Tooling to allow rapid new table development!

Plugins define config input and logger output

Complicated and resource intensive C++ build

294 C++11 sources: 5-9mins

Performance, end to end, and regression testing

Static and dynamic analysis

Kernel extensions including unsafe stress tests

Snowflake build requirements

Must build public code, and have public UI

Must build C++11 and cannot build in TravisCI (memory)

Must support various OS X versions

Must be trusted to produce packages automatically

Must have “some” Internet access

Distributing software is challenging

Static libraries distributed without -fPIC

x86_64 esoteric instruction set features

Outdated and vulnerable shared library support

Memory leaks in OS and open source C++ libraries

What if you cannot avoid leaking?

Case study: radar:19966048
SecDERItemCopyOIDDecimalRepresentation

Use a worker/watchdog model with strict RSS limits

```
SELECT i.pid, i.version, p.resident_size, p.user_time, p.system_time, uptime.total_seconds
FROM osquery_info i, processes p, uptime WHERE p.pid = i.pid;
+-------+-------------------+---------------+-----------+-------------+---------------+
| pid   | version           | resident_size | user_time | system_time | total_seconds |
+-------+-------------------+---------------+-----------+-------------+---------------+
| 94768 | 1.8.2-157-gf21f931 | 12754944      | 64        | 60          | 306872        |
+-------+-------------------+---------------+-----------+-------------+---------------+
```

Platforms and Distributions

OSX 10.9/10.10/10.11/10.12

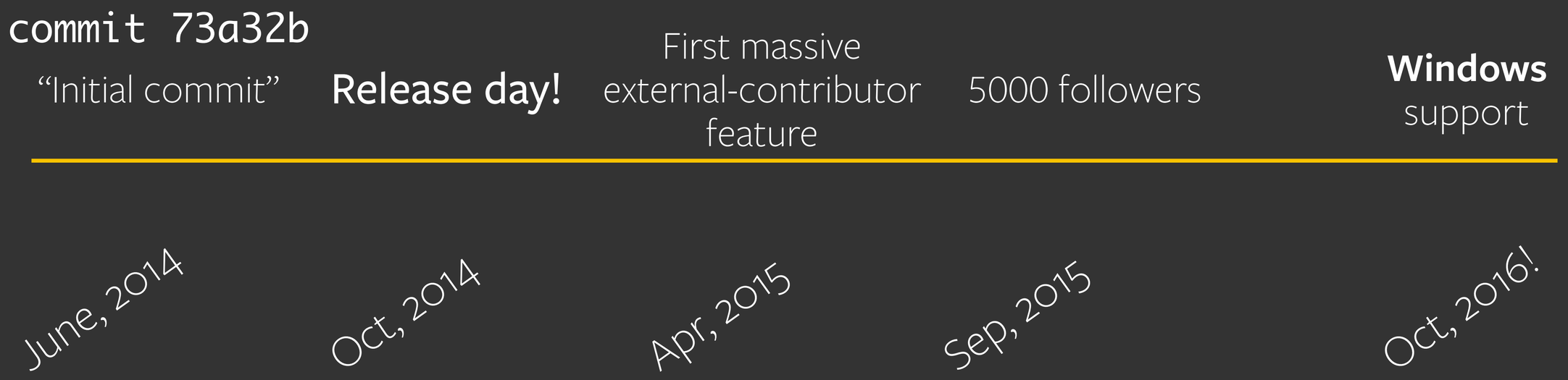
Ubuntu 12/14/16

CentOS 6/7, RHEL 6/7

FreeBSD 10

Windows 8/10/2008/2012

Two years of open source activity



7,306 followers

3,296 commits

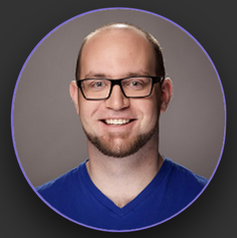
119 contributors

1 of hundreds of repos

Facebook Bug Bounty

Give it a try

https://osquery.io



Teddy Reed
Facebook
@teddyreedv

Nick Anderson
Facebook
@PoppySeedPlehzr

Michael McGrew
Facebook
@mtmcgrew

Sereyvathana Ty
Facebook
@sereyvathanaty

Mitchell Grenier
Facebook
@jedi22