

Unpacking the BADBOX Botnet with Censys

 censys.com/unpacking-the-badbox-botnet/

Executive Summary: BADBOX is a newly discovered botnet targeting both off-brand and well-known Android devices—often with malware that potentially came pre-installed from the factory or further down in the supply chain. Over 190,000 infected devices have been observed so far, including higher-end models like Yandex 4K QLED TVs. Using Censys, I identified a suspicious SSL/TLS certificate common to BADBOX infrastructure, revealing five IPs and numerous domains, all using the same certificate and SSH host key. This strongly indicates a single actor controlling a templated environment. The sheer scale and stealthy nature of BADBOX underscore the critical need to monitor supply chain integrity and network traffic.

I've been watching this emerging threat for a while, and on the surface, it sounds like just another Android malware campaign. The twist? BADBOX often comes baked into the firmware, so people are unboxing new devices that are already compromised before they even join a network. Researchers from BitSight [recently highlighted](#) the huge number of devices communicating with BADBOX servers, suggesting a full-blown supply chain compromise that goes well beyond a typical sideloaded malware incident. Below, I'll walk you through how I used Censys to track the certificate in question and map out the associated IPs and domains.

This scale piqued my curiosity—particularly the part about a common certificate that's been spotted in the wild. Armed with this bit of intel on the certificate's issuer DN, I turned to the [Censys Internet Intelligence Platform](#) to see if I could track down any additional evidence. The issuer DN in question is: "C=65, ST=singapore, L=singapore, O=singapre, OU=sall, CN=sae" which I converted into the following Certificate query to find the exact certificate used by BADBOX operators.

There was a single result that matched that criteria, which is a strong indicator of a single entity (or a small group) behind the widespread malware injection.

sae • CERTIFICATE AS OF SEP 09, 2023 | 05:48 UTC

never-trustedself-signedunexpireduntrusted

Issuer

C=65, ST=singapore, L=singapore, O=singapre, OU=sall, CN=sae

Validity Period

DEC 12, 2020 | DEC 10, 2030

Browser Trust

Untrusted x

All Names

sae

🔍 MATCHED FIELDS

cert.parsed.issuer_dn C=65, ST=singapore, L=singapore, O=singapre, OU=sall, CN=sae

This made me curious about what hosts this certificate is presented on so I entered the pivot menu.

< Results

sae **CERTIFICATE**

AS OF SEP 09, 2023 | 05:48 UTC

Download PEM **Pivots**
 never-trusted self-signed unexpired untrusted

Summary

Subject DN C=65, ST=singapore, L=singapore, O=singapre, OU=sall, CN=sae
Issuer DN C=65, ST=singapore, L=singapore, O=singapre, OU=sall, CN=sae
Serial Number Hex Decimal 0x90452e0ba6bccfa0
Validity Period DEC 12, 2020 | DEC 10, 2030 (3650 days, 0:00:00)
Fingerprint SHA-256 SHA-1 61609d6
All Names saee

Trust and Revocation

| Browser | Status | Trusted Path | Revoked | Expired |
|---------|---------------|--------------|---------|---------|
| Chrome | x Self-Signed | No | No | No |
| Edge | x Self-Signed | No | No | No |
| Firefox | x Self-Signed | No | No | No |
| Safari | x Self-Signed | No | No | No |

Certificate Trust ZLint Certificate Transparency Raw Data

Certificate Details

PUBLIC KEY

Key Type 1024-bit RSA, e = 65,537 **Insecure**
Modulus b4:19:9b:ae:14:d0:4e:47:7c:d7:36:5a:13:65:a5:e6:4d:96:59:b9:05:21:1a:5b:13:ca:c4:f5:5a:63:91:ab:9b:dc:7a:48:ce:... more
SPKI SHA-256 08f095706e19b31efde46cfaa14d19dfef407eb137032a66033470a2067f41fa

SIGNATURE

Algorithm SHA256-RSA (1.2.840.113549.1.1.11)
Signature a9:c5:79:51:4a:54:e6:4c:d5:64:ae:66:dc:56:0d:88:3e:40:d7:42:7f:75:56:4c:a8:43:75:40:97:13:92:8a:90:d8:78:d9:78:... more

Download PEM

Pivots

RELATED CERTIFICATES

With the same identity (key + subject)

With the same public key

With the same serial number

With the same names

CERTIFICATE TRANSPARENCY

Associated Pre-certificates

WHAT'S USING THIS CERTIFICATE?

Hosts

This pivot produced the following query, which searches for the certificate's SHA-256 fingerprint.

The screenshot shows a search interface with a search bar containing the query: `host.services.tls.fingerprint_sha256 = "61609d67762922a390bf4c5ccc2b5ed43c1980a6777a0152e9a49c5b96d0d623"`. The search results are displayed in a list format, showing five IP addresses that match the query. Each result includes the IP address, the host name, and a list of services running on the host. The services listed are: 22 / SSH, 3307 / MYSQL, 443 / HTTP, 80 / HTTP, 8081 / HTTP, 8082 / HTTP, and 8085 / HTTP. The location for all results is Singapore, (SG). The network is AKAMAI-LINODE-AP Akamai Connected Cloud (63949). The matched fields are: `host.services.tls.fingerprint_sha256 61609d67762922a390bf4c5ccc2b5ed43c1980a6777a0152e9a49c5b96d0d623`.

| IP Address | Host Name | Services | Location | Network |
|-----------------|------------------------------------------|--------------------------------------------------------------------------------------|-----------------|-------------------------------------------------|
| 192.46.227.25 | 192-46-227-25.ip.linodeusercontent.com | 22 / SSH, 3307 / MYSQL, 443 / HTTP, 80 / HTTP, 8081 / HTTP, 8082 / HTTP, 8085 / HTTP | Singapore, (SG) | AKAMAI-LINODE-AP Akamai Connected Cloud (63949) |
| 172.104.186.191 | 172-104-186-191.ip.linodeusercontent.com | 22 / SSH, 3307 / MYSQL, 443 / HTTP, 80 / HTTP | Singapore, (SG) | AKAMAI-LINODE-AP Akamai Connected Cloud (63949) |
| 143.42.75.145 | 143-42-75-145.ip.linodeusercontent.com | 22 / SSH, 443 / HTTP, 80 / HTTP, 8085 / HTTP | Singapore, (SG) | AKAMAI-LINODE-AP Akamai Connected Cloud (63949) |
| 139.162.40.221 | 139-162-40-221.ip.linodeusercontent.com | 22 / SSH, 3307 / MYSQL, 443 / HTTP, 80 / HTTP | Singapore, (SG) | AKAMAI-LINODE-AP Akamai Connected Cloud (63949) |

This returned five IP addresses that are presenting that certificate, all from Singapore and all from the Akamai ASN. I was curious what other attributes they share and I noticed that they all have port 22 SSH open. Here is one of those services.

The screenshot shows the details of an SSH 22 / TCP service. The page includes a header with the service name, last observed time, and UTC offset. The details section shows the host key algorithm and fingerprint, and the negotiated key exchange, symmetric cipher, and MAC.

| Section | Attribute | Value |
|------------|------------------|------------------------------------------------------------------|
| HOST KEY | Algorithm | ecdsa-sha2-nistp256 |
| | Fingerprint | a885b892e4820b90fd05e45eda6bdd5983170cba6da23fb3610ed1a61726bd14 |
| NEGOTIATED | Key Exchange | curve25519-sha256@libssh.org |
| | Symmetric Cipher | aes128-ctr aes128-ctr |
| | MAC | hmac-sha2-256 hmac-sha2-256 |

To track if the same SSH Host Keys are used, we can do a report on the Host Key Fingerprint field “host.services.ssh.server_host_key.fingerprint_sha256”. To do a report, click the “Report Builder” tab.

The screenshot shows the Censys Report Builder interface. At the top, there are two tabs: "Search Results" and "Report Builder", with the latter being selected and highlighted with an orange box. Below the tabs, there is a search bar containing the query: `host.services.tls.fingerprint_sha256 = "61609d67762922a390bf4c5ccc2b5ed43c1980a6777a0152e9a49c5b96d0d623"`. To the right of the search bar is a "Search" button. Below the search bar, there are two sub-tabs: "Search Results" and "Report Builder", with the latter being selected. The main content area is titled "Report on Hosts, Certificates, or Web Properties" and contains a description of the tool. Below this, there is a "Breakdown Field" dropdown menu set to `host.services.ssh.server_host_key.fingerprint_sha256`. To the right of the dropdown is a "Buckets" dropdown menu set to "# 50". To the right of the "Buckets" dropdown is a "Generate Report" button. Below the dropdowns, there are two buttons: "Table" and "JSON", with "Table" being selected. Below the buttons, there is a table showing the results of the report. The table has three columns: "host.services.ssh.server_host_key.fingerprint_sha256", "Count", and "%". The first row shows the fingerprint `a885b892e4820b90fd05e45eda6bdd5983170cba6da23fb3610ed1a61726bd14` with a count of 5 and 100.00%. The second row shows "Remaining Results" with a count of 0 and 0.00%. The third row shows "Total" with a count of 5 and 100.00%.

| host.services.ssh.server_host_key.fingerprint_sha256 | Count | % |
|----------------------------------------------------------------------------------|-------|---------|
| a885b892e4820b90fd05e45eda6bdd5983170cba6da23fb3610ed1a61726bd14 | 5 | 100.00% |
| Remaining Results | 0 | 0.00% |
| Total | 5 | 100.00% |

As you can see, all five IPs share the same SSH Host Key suggesting that these instances were templated. By clicking on the report's table I can pivot into that query.

Which I would clean up to be the following query:

However, I was also interested in the number of domains that also present this certificate.

Search Results Report Builder

Search: Search Help AH

ASSET TYPES

- Hosts 0
- Certificates 0
- Web Properties 25

APPLICATIONS

- HTTP 25

SOFTWARE VENDORS

- f5 25

SOFTWARE PRODUCTS

- nginx 25

RESULTS: 25 • DURATION: 1.18s

ztword.com: 443 • WEB PROPERTY AS OF JAN 20, 2025 07:58 UTC

| | | |
|-----------------------------|------------------------------------------------------------------|------------|
| HTML Title | Welcome to CentOS | 1 Endpoint |
| Browser Trust | Untrusted x | 443 / HTTP |
| Software | Nginx 1.20.1 | |
| MATCHED FIELDS | | |
| web.cert.fingerprint_sha256 | 61609d67762922a390bf4c5ccc2b5ed43c1980a6777a0152e9a49c5b96d0d623 | |

www.yydsmd.com: 443 • WEB PROPERTY AS OF JAN 05, 2025 10:45 UTC

| | | |
|-----------------------------|------------------------------------------------------------------|------------|
| HTML Title | Welcome to CentOS | 1 Endpoint |
| Browser Trust | Untrusted x | 443 / HTTP |
| Software | Nginx 1.20.1 | |
| MATCHED FIELDS | | |
| web.cert.fingerprint_sha256 | 61609d67762922a390bf4c5ccc2b5ed43c1980a6777a0152e9a49c5b96d0d623 | |

www.yydsmb.com: 443 • WEB PROPERTY AS OF JAN 16, 2025 21:16 UTC

| | | |
|-----------------------------|------------------------------------------------------------------|------------|
| HTML Title | Welcome to CentOS | 1 Endpoint |
| Browser Trust | Untrusted x | 443 / HTTP |
| Software | Nginx 1.20.1 | |
| MATCHED FIELDS | | |
| web.cert.fingerprint_sha256 | 61609d67762922a390bf4c5ccc2b5ed43c1980a6777a0152e9a49c5b96d0d623 | |

www.mtcpuouo.com: 443 • WEB PROPERTY AS OF JAN 17, 2025 01:55 UTC

| | | |
|-----------------------------|------------------------------------------------------------------|------------|
| HTML Title | Welcome to CentOS | 1 Endpoint |
| Browser Trust | Untrusted x | 443 / HTTP |
| Software | Nginx 1.20.1 | |
| MATCHED FIELDS | | |
| web.cert.fingerprint_sha256 | 61609d67762922a390bf4c5ccc2b5ed43c1980a6777a0152e9a49c5b96d0d623 | |

Interestingly enough, all 25 appear to be running nginx 1.20.1 on CentOS. From here I could either make a collection to track all of these indicators or simply extract the current instances. Below is the final query with all the above indicators

host.services.tls.fingerprint_sha256 =
 "61609d67762922a390bf4c5ccc2b5ed43c1980a6777a0152e9a49c5b96d0d623"

or host.services.ssh.server_host_key.fingerprint_sha256 =
 "a885b892e4820b90fd05e45eda6bdd5983170cba6da23fb3610ed1a61726bd14"

or web.cert.fingerprint_sha256 =
 "61609d67762922a390bf4c5ccc2b5ed43c1980a6777a0152e9a49c5b96d0d623"

Indicators

IPs

139.162.36[.]224

139.162.40[.]221

143.42.75[.]145

172.104.186[.]191

192.46.227[.]25

172.104.178[.]158

Domains

bluefish[.]work

www.bluefish[.]work

cool.hbmc[.]net

giddy[.]cc

www.giddy[.]cc

jolted[.]vip

joyfulxx[.]com

msohu[.]shop

www.msohu[.]shop

mtcpuouo[.]com

www.mtcpuouo[.]com

pasiont[.]com

sg100.idcloudhost[.]com

www.yydsmb[.]com

www.yydsmd[.]com

ztword[.]com tvsnapp[.]com

pixelscast[.]com

swiftcode[.]work

old.1ztop[.]work

cast.jutux[.]work

home.1ztop[.]work

www.jolted[.]vip

Author

Aidan Holland

Security Researcher



Aidan is a Security Researcher on the Research team working to use our data to enrich the workflows of security professionals everywhere. Aidan specializes in open-source development and cybersecurity engineering.