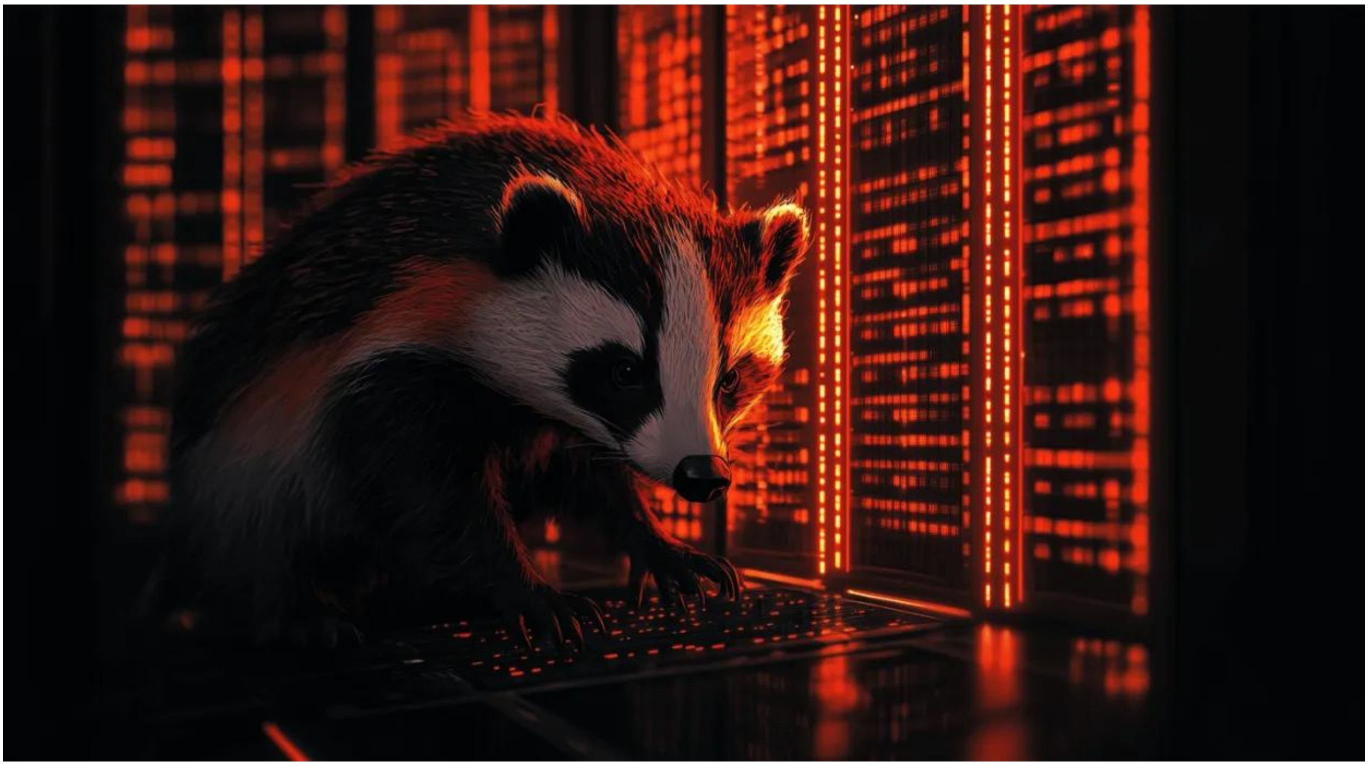
Chasing Badgers: Infrastructure Hunting with OpenCTI and Shodan



By Mario Rojas, Senior Security Researcher, ORYXLABS

Note: This post departs from our usual analytical format and adopts a more exploratory, field-journal style. The shift reflects the investigative and hands-on nature of infrastructure hunting, where agility and narrative context are essential to conveying process and insight.

A few days ago, I was reading Symantec's report on cyber operations in the Middle East and something in the Seedworm (a.k.a. MuddyWater) section caught my eye: this Advanced Persistent Threat (APT) is using Brute Ratel C4 (BRc4) in its operations.

There have been numerous reports of different threat groups deploying Brute Ratel C4 (BRc4). However, this is the first time we've seen public confirmation of MuddyWater leveraging it. The group has a history of abusing legitimate tools in its operations, so its decision to add BRc4 to its arsenal was worth a closer look. Here we share some of the passive OSINT techniques used to track "Badger" infrastructure (Brute Ratel's term for its endpoint agents) so threat hunters can proactively identify and block it.

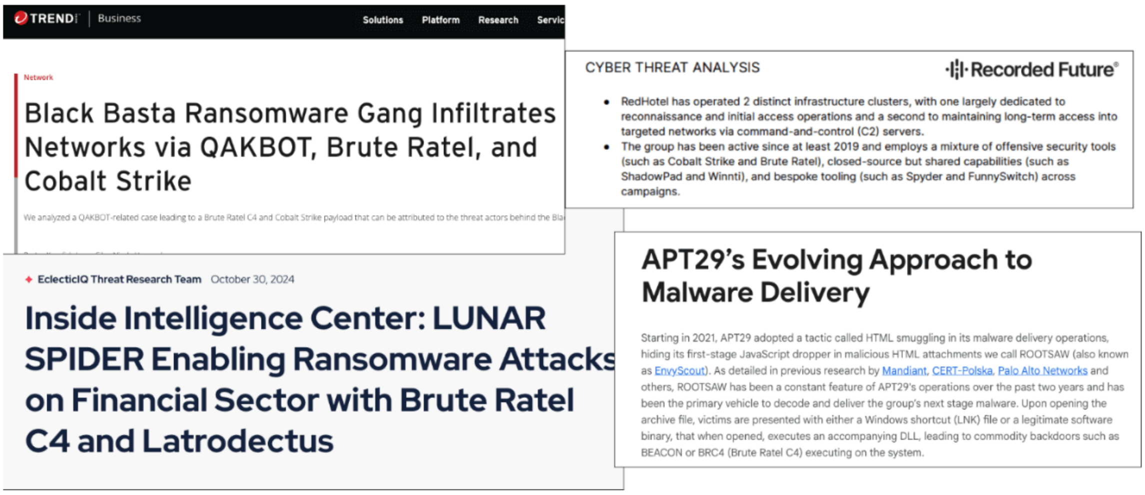
Who is MuddyWater?

MuddyWater (a.k.a. Seedworm, Earth Vetala, MERCURY, Static Kitten, TEMP.Zagros, Mango Sandstorm, TA450) is a cyber espionage group publicly attributed to Iran's Ministry of Intelligence and Security (MOIS). Active since at least 2017, the group has targeted government, telecom, energy, and other critical sectors across the Middle East, Europe and North America.

Its operations often involve exploiting known vulnerabilities for initial access, followed by the use of legitimate tools such as remote administration utilities and tunnelling software to blend in with normal network activity. The APT’s toolset has included custom malware, open-source frameworks, and now BRc4.

A quick note on Brute Ratel

BRc4 is a legitimate red-teaming tool built for stealth. It employs techniques such as memory encryption [T1027], userland hook removal [T1562.006], and sleep masking [T1055.002] to avoid detection, features that make it attractive to both red teams and threat actors.



Network

Black Basta Ransomware Gang Infiltrates Networks via QAKBOT, Brute Ratel, and Cobalt Strike

We analyzed a QAKBOT-related case leading to a Brute Ratel C4 and Cobalt Strike payload that can be attributed to the threat actors behind the Bla

CYBER THREAT ANALYSIS

Recorded Future

- RedHotel has operated 2 distinct infrastructure clusters, with one largely dedicated to reconnaissance and initial access operations and a second to maintaining long-term access into targeted networks via command-and-control (C2) servers.
- The group has been active since at least 2019 and employs a mixture of offensive security tools (such as Cobalt Strike and Brute Ratel), closed-source but shared capabilities (such as ShadowPad and Winnti), and bespoke tooling (such as Spyder and FunnySwitch) across campaigns.

✦ EclecticIQ Threat Research Team October 30, 2024

Inside Intelligence Center: LUNAR SPIDER Enabling Ransomware Attacks on Financial Sector with Brute Ratel C4 and Latrodectus

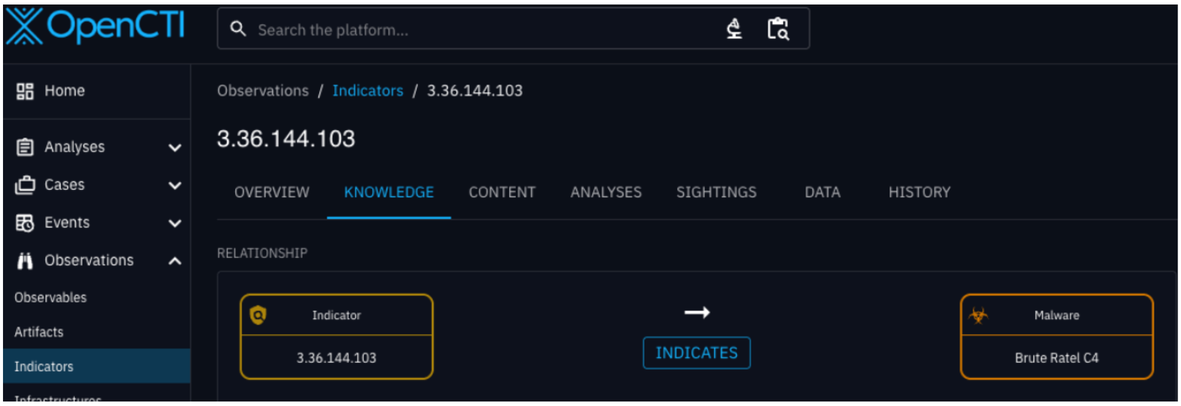
APT29’s Evolving Approach to Malware Delivery

Starting in 2021, APT29 adopted a tactic called HTML smuggling in its malware delivery operations, hiding its first-stage JavaScript dropper in malicious HTML attachments we call ROOTSAW (also known as EnvyScout). As detailed in previous research by Mandiant, CERT-Polska, Palo Alto Networks and others, ROOTSAW has been a constant feature of APT29's operations over the past two years and has been the primary vehicle to decode and deliver the group’s next stage malware. Upon opening the archive file, victims are presented with either a Windows shortcut (LNK) file or a legitimate software binary, that when opened, executes an accompanying DLL, leading to commodity backdoors such as BEACON or BRC4 (Brute Ratel C4) executing on the system.

Since a cracked version was leaked in 2022, it has been abused in campaigns by threat groups such as Black Basta and LUNAR SPIDER, and in espionage operations tied to regions including the Middle East. These groups favour tools like Brute Ratel as their stealthy payloads, which are often hard to catch. Its infrastructure, though? Not as much.

Where the hunt begins

I decided to start the “hunt” by identifying a known BRc4 command-and-control (C2) instance from our OpenCTI platform, then used resources such as Shodan and VirusTotal to track additional infrastructure.



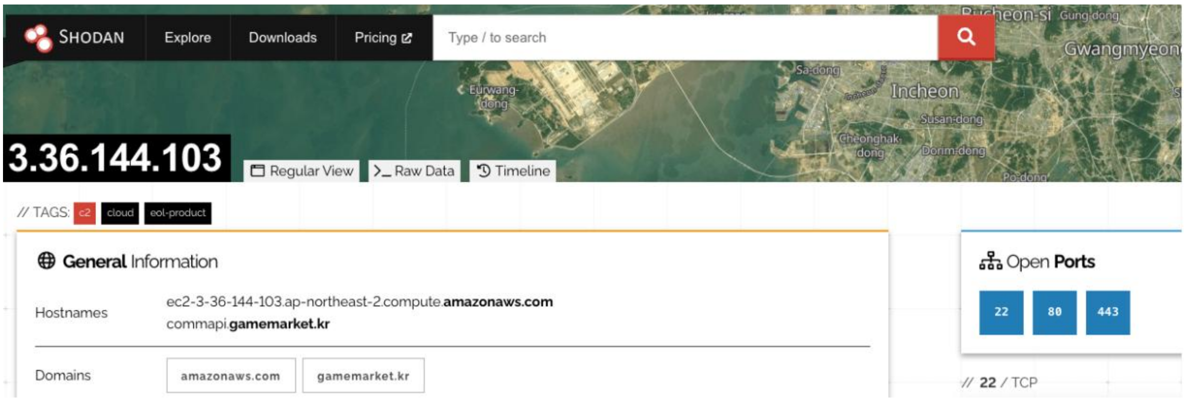
Observations / Indicators / 3.36.144.103

3.36.144.103

RELATIONSHIP

Indicator 3.36.144.103 → INDICATES → Malware Brute Ratel C4

After locating the starting point (3[.]36[.]144[.]103), I turned to Shodan, which revealed a few open ports: 22 (SSH), 80 (HTTP), and 443 (HTTPS), and had already flagged the host as a C2.



3.36.144.103

// TAGS: c2 cloud eol-product

General Information

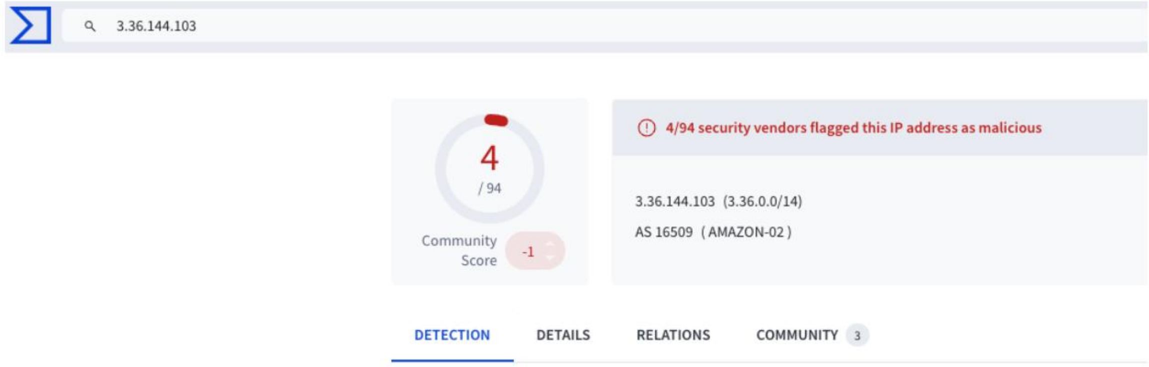
Hostnames	ec2-3-36-144-103.ap-northeast-2.compute.amazonaws.com commapi.gamemarket.kr
Domains	amazonaws.com gamemarket.kr

Open Ports

22 80 443

// 22 / TCP

Then, I moved to check VirusTotal for information on this host. Surprisingly, it showed only a few detections (4/94).



3.36.144.103

4 / 94

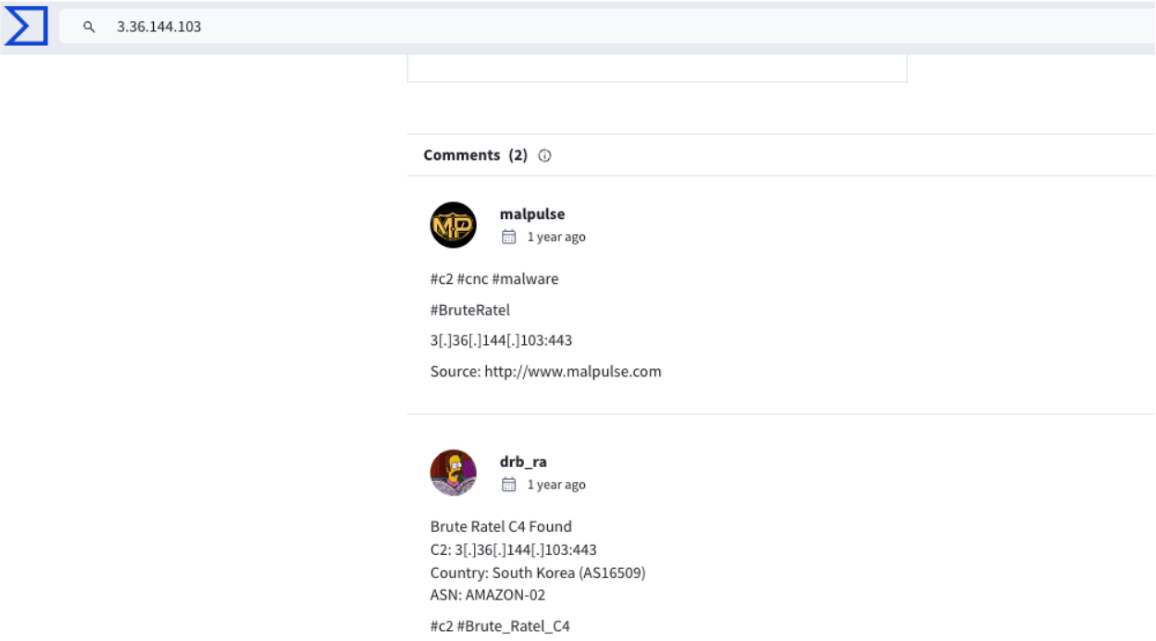
Community Score -1

4/94 security vendors flagged this IP address as malicious

3.36.144.103 (3.36.0.0/14)

AS 16509 (AMAZON-02)

Looking at the “Community” tab confirmed the presence of Brute Ratel on this host, which, based on the timestamp, has been the case for at least a year. That is Persistence…



Comments (2)

malpulse
1 year ago

#c2 #cnc #malware

#BruteRatel

3[.]36[.]144[.]103:443

Source: http://www.malpulse.com

drb_ra
1 year ago

Brute Ratel C4 Found
C2: 3[.]36[.]144[.]103:443
Country: South Korea (AS16509)
ASN: AMAZON-02

#c2 #Brute_Ratel_C4

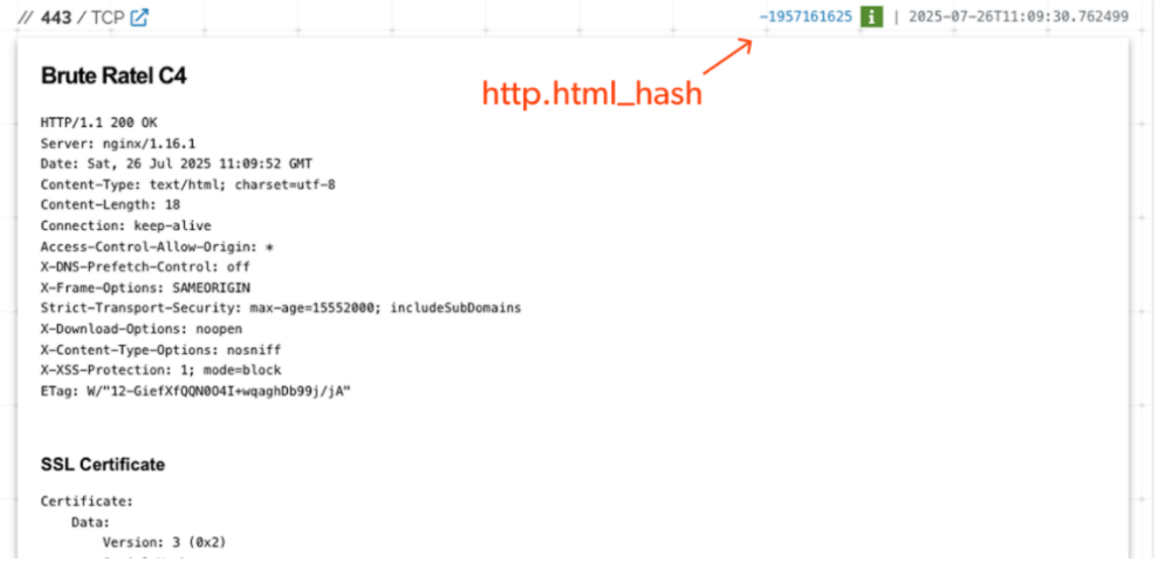
The pivots

In OSINT, pivoting refers to using one piece of information to uncover related data, thereby effectively expanding the investigation beyond the initial data point. It's time to jump back to Shodan and continue hunting. Among the myriad of data Shodan collects about the hosts, there are a few "identifiers" we can use as pivot points:

- HTTP Headers
- HTML Hash
- Certificate Details
- JARM fingerprint

HTML Hash

I decided to start with HTML hash - Shodan's fingerprint of the actual HTML response body from the server. This is particularly useful when the same C2 toolkit serves a static or templated page. You'll find it under the 443 tab.



// 443 / TCP

-1957161625 | 2025-07-26T11:09:30.762499

http.html_hash

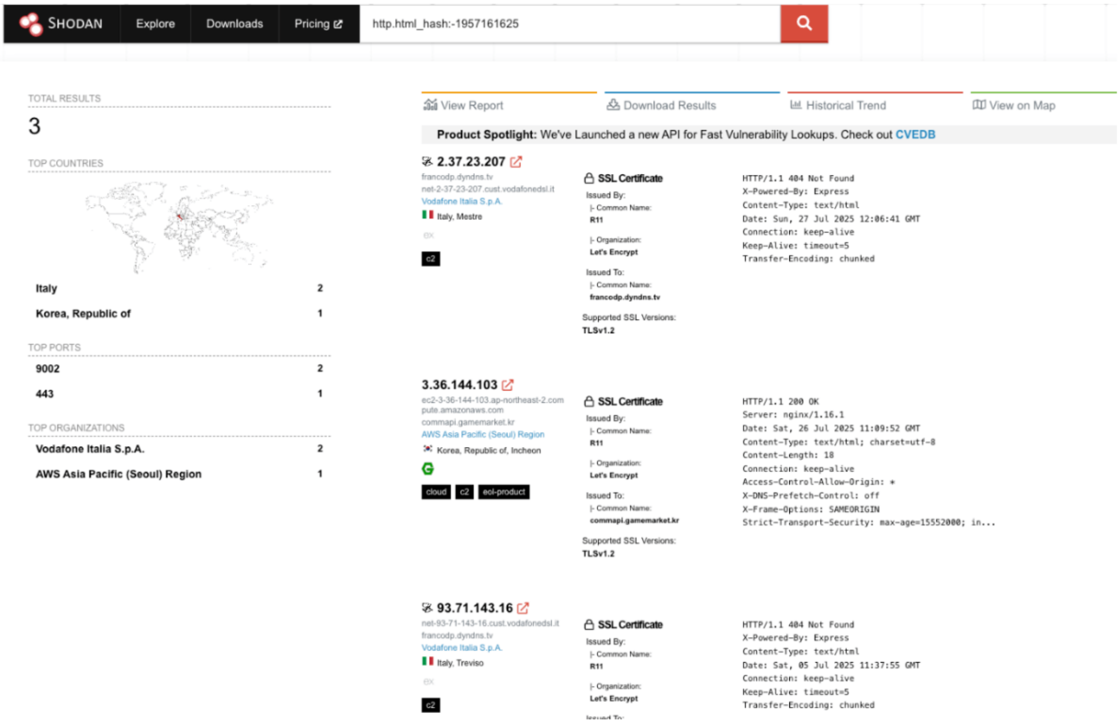
Brute Ratel C4

```
HTTP/1.1 200 OK
Server: nginx/1.16.1
Date: Sat, 26 Jul 2025 11:09:52 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 18
Connection: keep-alive
Access-Control-Allow-Origin: *
X-DNS-Prefetch-Control: off
X-Frame-Options: SAMEORIGIN
Strict-Transport-Security: max-age=15552000; includeSubDomains
X-Download-Options: noopen
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
ETag: W/"12-GiefXfQQN004I+wqaghDb99j/jA"
```

SSL Certificate

```
Certificate:
    Data:
        Version: 3 (0x2)
```

Clicking on it runs a new query using the http.html_hash filter, which, in our case, returned two more Badgers. Both were already tagged as C2, so no surprises there.



TOTAL RESULTS

3

TOP COUNTRIES

Italy	2
Korea, Republic of	1

TOP PORTS

9002	2
443	1

TOP ORGANIZATIONS

Vodafone Italia S.p.A.	2
AWS Asia Pacific (Seoul) Region	1

Product Spotlight: We've Launched a new API for Fast Vulnerability Lookups. Check out CVEDB

2.37.23.207
francodp.dyndns.tv
net-2-37-23-207.cust.vodafonedsl.it
Vodafone Italia S.p.A.
Italy, Mestre
c2

SSL Certificate
Issued By:
|- Common Name: R11
|- Organization: Let's Encrypt
Issued To:
|- Common Name: francodp.dyndns.tv
Supported SSL Versions: TLSv1.2

```
HTTP/1.1 404 Not Found
X-Powered-By: Express
Content-Type: text/html
Date: Sun, 27 Jul 2025 12:06:41 GMT
Connection: keep-alive
Keep-Alive: timeout=5
Transfer-Encoding: chunked
```

3.36.144.103
ec2-3-36-144-103.ap-northeast-2.compute.amazonaws.com
commapi.gamemarket.kr
AWS Asia Pacific (Seoul) Region
Korea, Republic of, Incheon
cloud c2 eol-product

SSL Certificate
Issued By:
|- Common Name: R11
|- Organization: Let's Encrypt
Issued To:
|- Common Name: commapi.gamemarket.kr
Supported SSL Versions: TLSv1.2

```
HTTP/1.1 200 OK
Server: nginx/1.16.1
Date: Sat, 26 Jul 2025 11:09:52 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 18
Connection: keep-alive
Access-Control-Allow-Origin: *
X-DNS-Prefetch-Control: off
X-Frame-Options: SAMEORIGIN
Strict-Transport-Security: max-age=15552000; in...
```

93.71.143.16
net-93-71-143-16.cust.vodafonedsl.it
francodp.dyndns.tv
Vodafone Italia S.p.A.
Italy, Treviso
c2

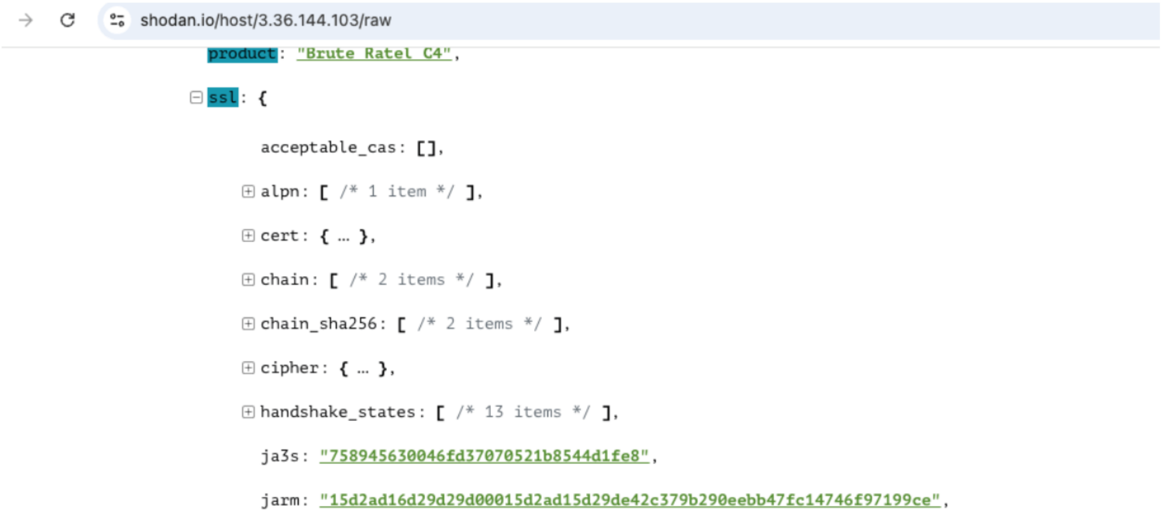
SSL Certificate
Issued By:
|- Common Name: R11
|- Organization: Let's Encrypt
Issued To:

```
HTTP/1.1 404 Not Found
X-Powered-By: Express
Content-Type: text/html
Date: Sat, 05 Jul 2025 11:37:55 GMT
Connection: keep-alive
Keep-Alive: timeout=5
Transfer-Encoding: chunked
```

Even though this was useful, I was hoping to find infrastructure that hadn’t been flagged yet, so I continued exploring other pivot points.

JARM

JARM is a TLS fingerprinting technique that hashes a server’s behaviour across multiple handshake attempts. The idea is that similar configurations generate similar hashes; this is especially true for tools such as Brute Ratel, which use embedded Go-based HTTP servers.

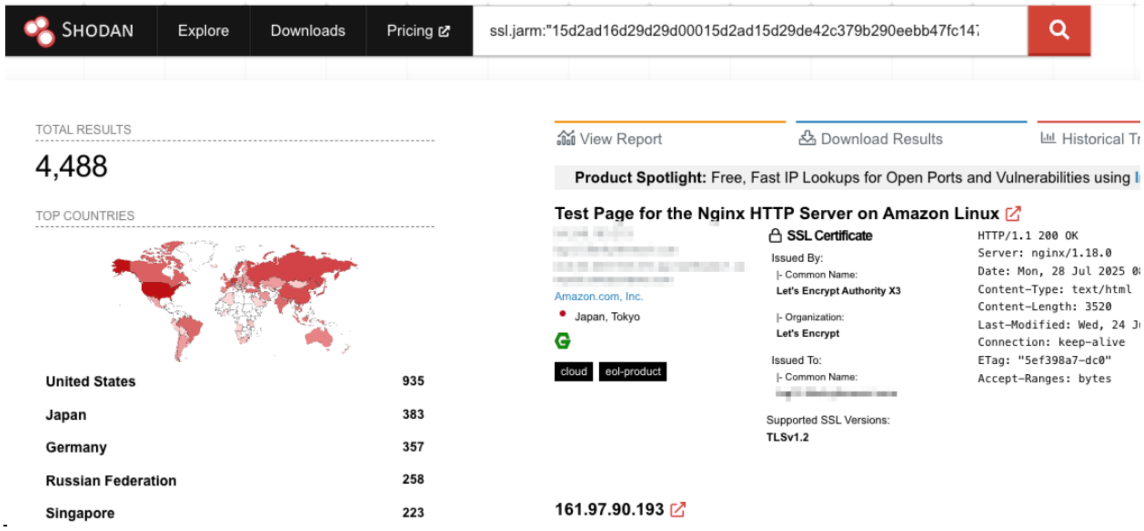


shodan.io/host/3.36.144.103/raw

```
    product: "Brute Ratel C4",
  ⊟ ssl: {
        acceptable_cas: [],
      ⊞ alpn: [ /* 1 item */ ],
      ⊞ cert: { … },
      ⊞ chain: [ /* 2 items */ ],
      ⊞ chain_sha256: [ /* 2 items */ ],
      ⊞ cipher: { … },
      ⊞ handshake_states: [ /* 13 items */ ],
        ja3s: "758945630046fd37070521b8544d1fe8",
        jarm: "15d2ad16d29d29d00015d2ad15d29de42c379b290eebb47fc14746f97199ce",
```

Shodan lists the JARM under raw data:

Clicking the JARM hash will run a new query.



ssl.jarm:"15d2ad16d29d29d00015d2ad15d29de42c379b290eebb47fc147

TOTAL RESULTS

4,488

TOP COUNTRIES

United States	935
Japan	383
Germany	357
Russian Federation	258
Singapore	223

Product Spotlight: Free, Fast IP Lookups for Open Ports and Vulnerabilities using I

Test Page for the Nginx HTTP Server on Amazon Linux

[redacted]

Amazon.com, Inc.

Japan, Tokyo

cloud eol-product

SSL Certificate

Issued By:
|- Common Name:
Let's Encrypt Authority X3

|- Organization:
Let's Encrypt

Issued To:
|- Common Name:
[redacted]

Supported SSL Versions:
TLSv1.2

```
HTTP/1.1 200 OK
Server: nginx/1.18.0
Date: Mon, 28 Jul 2025 0
Content-Type: text/html
Content-Length: 3520
Last-Modified: Wed, 24 J
Connection: keep-alive
ETag: "5ef398a7-dc0"
Accept-Ranges: bytes
```

161.97.90.193

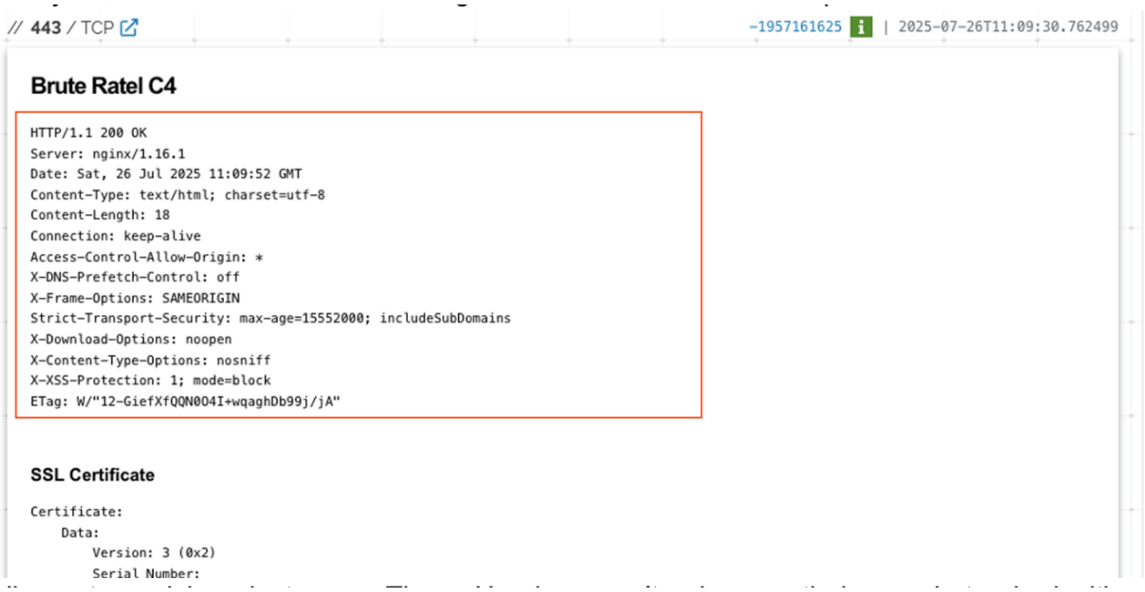
This hash returned close to 4,500 results. As you can see, JARM alone is too noisy, but when combined with other data points, it can help filter down candidate infrastructures.

HTTP Headers

From the same host, I pulled the HTTP headers. These can be surprisingly helpful. Brute Ratel often serves minimal content (same length, same type) with no real templating or randomness.

Here is what this one returned:

Instead of using the full range of Headers, I decided to take a subset that has worked in previous hunts:



// 443 / TCP

-1957161625 | 2025-07-26T11:09:30.762499

Brute Ratel C4

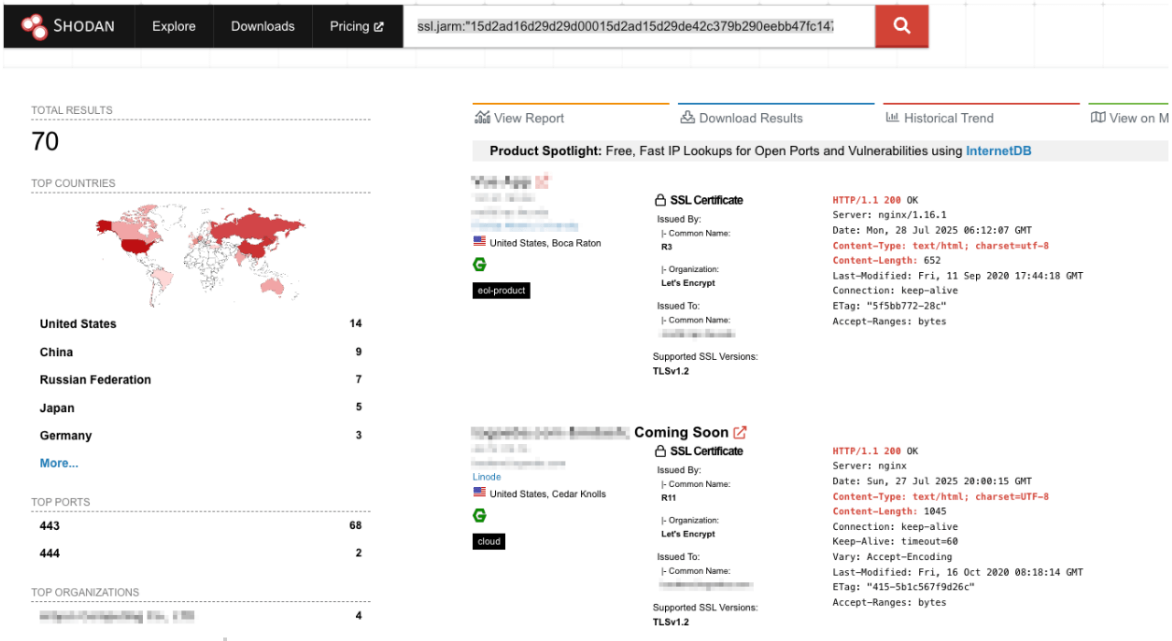
```
HTTP/1.1 200 OK
Server: nginx/1.16.1
Date: Sat, 26 Jul 2025 11:09:52 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 18
Connection: keep-alive
Access-Control-Allow-Origin: *
X-DNS-Prefetch-Control: off
X-Frame-Options: SAMEORIGIN
Strict-Transport-Security: max-age=15552000; includeSubDomains
X-Download-Options: noopen
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
ETag: W/"12-GiefXfQQN004I+wqaghDb99j/jA"
```

SSL Certificate

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
```

- HTTP/1.1 200 OK
- Content-Type: text/html; charset=utf-8
- Content-Length: 18

That content length is interesting. I’ve seen multiple C2 servers return responses with exactly 14 or 18 bytes, which is an uncommon length for an HTTPS 200 OK response and can be used to easily spot suspicious instances. These Headers aren’t unique on their own, but paired with a JARM, they help filter noise significantly.



ssl.jarm:"15d2ad16d29d29d00015d2ad15d29de42c379b290eebb47fc147

TOTAL RESULTS

70

TOP COUNTRIES

United States	14
China	9
Russian Federation	7
Japan	5
Germany	3
More...	

TOP PORTS

443	68
444	2

TOP ORGANIZATIONS

[redacted]	4

Product Spotlight: Free, Fast IP Lookups for Open Ports and Vulnerabilities using InternetDB

[redacted]
United States, Boca Raton
eol-product

SSL Certificate
Issued By:
|- Common Name:
R3
|- Organization:
Let's Encrypt
Issued To:
|- Common Name:
[redacted]
Supported SSL Versions:
TLSv1.2

```
HTTP/1.1 200 OK
Server: nginx/1.16.1
Date: Mon, 28 Jul 2025 06:12:07 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 652
Last-Modified: Fri, 11 Sep 2020 17:44:18 GMT
Connection: keep-alive
ETag: "5f5bb772-28c"
Accept-Ranges: bytes
```

[redacted] Coming Soon
Linode
United States, Cedar Knolls
cloud

SSL Certificate
Issued By:
|- Common Name:
R11
|- Organization:
Let's Encrypt
Issued To:
|- Common Name:
[redacted]
Supported SSL Versions:
TLSv1.2

```
HTTP/1.1 200 OK
Server: nginx
Date: Sun, 27 Jul 2025 20:00:15 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 1045
Connection: keep-alive
Keep-Alive: timeout=60
Vary: Accept-Encoding
Last-Modified: Fri, 16 Oct 2020 08:18:14 GMT
ETag: "415-5b1c567f9d26c"
Accept-Ranges: bytes
```

Here is the full search I used:

```
ssl.jarm:"15d2ad16d29d29d00015d2ad15d29de42c379b290eebb47fc14746f97199ce" HTTP/1.1 200 OK Content-Type: text/html; charset=utf-8 Content-Length: 18
```

This reduced potential Brute Ratel instances to around 70, a much better baseline than the

4,500 we started with. This is where combining pivot points really pays off, significantly improving the quality of your findings.

What is next?

As shown above, these simple pivots can help surface active C2 infrastructure that often evades automated detection. From here, you could:

- Cross-check in VirusTotal or ThreatFox to validate hits or identify overlaps with known campaigns.
- Look for reused certificates or domains across other hosts.
- Drop IPs into passive DNS tools to check for rotation patterns or staging behaviour.

These techniques aren’t exclusive to Brute Ratel; you can apply the same approach to other C2 frameworks such as Cobalt Strike, Sliver, or Mythic. Go and try it against a known IOC and see how far you can pivot.

I hope you found the walkthrough useful. Let me know if you discover other Badger-related infrastructure.

Note: All of this was passive. No interaction with suspected C2s.

Confirmed BRc4 C2s

Here is a short list of confirmed Brute Ratel servers found during this hunt:

Indicator	Type
3[.]36[.]144[.]103	IPv4
2[.]37[.]23[.]207	IPv4
93[.]71[.]143[.]16	IPv4
54[.]168[.]191[.]225	IPv4
54[.]65[.]227[.]196	IPv4
104[.]164[.]55[.]75	IPv4

Conclusion

Brute Ratel continues to appear in active campaigns, including those tied to targeted operations in the Middle East. Its infrastructure doesn’t always hide well, and passive hunting can expose far more than most feeds will reveal: starting from a single known C2, you can uncover dozens more using signals like HTML hashes, JARM fingerprints, and response headers.

If you’re defending a network, blocking these C2s before they hit makes a significant difference. You can also use solutions such as DNS FIREWALL to block outbound connections before they reach malicious infrastructure, while DISCOVERY will track credential leaks and domain abuse, helping identify this type of infrastructure and complete the picture.

Remember, there is always more intelligence to uncover.