ODIN

TLP:CLEAR

ODIN RESEARCH · MONOGRAPH 23 / 30 · POST-EXPLOITATION

Java Weaponizers and NTLM Traps
Relays, Workbenches and ICMP Tunnels

How JavaChains controllers, internet-facing Responder listeners and QUIC-over-ICMP shells split initial access, credential capture and covert egress

Two-week observation window ending 18 November 2026 · Prepared November 2026
Full Report · Public edition · Version 1.1

209
Endpoints across six tool families, on 175 distinct IPv4 hosts

23
Confirmed endpoints to block (11.0%): 16 JavaChains and 7 Responder, on 23 distinct addresses

42
JavaChains endpoints; 38 on port 8011

31 on 15
Responder endpoints on 15 hosts (2.07 endpoints per host), led by 7-port and 6-port mail hubs

What Is in This Report. 11 chapters and two appendices.

How JavaChains controllers, internet-facing Responder listeners and QUIC-over-ICMP shells split initial access, credential capture and covert egress

01	Executive Summary	04
02	Four Roles, Six Tool Families	07
03	Java Deserialization Weaponizers	10
04	Internet-Exposed Responder NTLM Traps	15
05	Covert ICMP Shells: the Termirror Workbench	18
06	Tunnels, Guardrails and Dual-Use Scanners	21
07	What Defenders Should Do	24
08	Detection Rules	26
09	Indicators	29
10	Scope and Limits	32
11	Sources	34
A	Indicators of compromise (23)	36
B	Detection rules (4 files)	37

IN SHORT

In a two-week observation window ending 18 November 2026, Odin by Brandefense mapped 209 endpoints on 175 hosts across six offensive tool families, plus three exposed staging workbenches. Java deserialization servers, internet-facing NTLM capture listeners and covert ICMP or TLS tunnels each play a separate role. Of the 209 endpoints, 23 are confirmed and ready to block. Block those 23 addresses, stop outbound SMB and WebDAV authentication at the perimeter, and inspect outbound ICMP Echo traffic for tunneled QUIC frames.

01

CHAPTER 01

Executive Summary

IN SHORT

In a two-week observation window ending 18 November 2026, Odin by Brandefense mapped 209 endpoints on 175 hosts across six offensive tool families, plus three exposed staging workbenches. Java deserialization servers, internet-facing NTLM capture listeners and covert ICMP or TLS tunnels each play a separate role. Of the 209 endpoints, 23 are confirmed and ready to block. Block those 23 addresses, stop outbound SMB and WebDAV authentication at the perimeter, and inspect outbound ICMP Echo traffic for tunneled QUIC frames.

INSIDE THIS CHAPTER

Three Decisions for Leaders 04

The Study in Numbers 04

01 Executive Summary

Attackers rarely rely on one protocol. One tool delivers the first Java exploit. A second tricks Windows hosts into sending their login handshakes to the internet. A third keeps a shell alive through a firewall that blocks normal outbound traffic. This report maps the servers behind each of these roles and the exposed workbenches where operators prepare them.

1. **JavaChains deserialization servers cluster on port 8011 in Chinese clouds.**
   Odin found 42 JavaChains endpoints on 41 hosts; 38 of them (90.5%) listen on TCP 8011. Sixteen are confirmed and sit on dedicated cloud servers. Two exposed workbenches add a large out-of-band exploit log and a RuoYi SnakeYAML payload builder.

2. **Responder NTLM capture listeners sit on public cloud servers.**
   Responder is normally an internal network tool. Here, 31 endpoints on 15 hosts expose it to the internet on web, remote desktop, SIP and mail ports. Seven listeners on DigitalOcean, Namecheap, BL Networks and Blackfoot Telephone are confirmed.

3. **A Termirror workbench hides QUIC shells inside ICMP Echo packets.**
   Scripts on the workbench run eight parallel shell streams and a 4 MB file transfer over ping traffic. A small relay falls back from TCP to UDP when ICMP is dropped. Ligolo-ng and RedWarden add TLS tunnels and C2 guardrails.

Three Decisions for Leaders

1. **Block the 23 confirmed servers, and only those.** Block the 16 JavaChains and 7 Responder addresses. Keep dual-use scanners, dynamic telecom addresses, suspected tunnels and candidates on time-bound watchlists.
2. **Stop Windows authentication from leaving the network.** Deny outbound SMB on ports 139 and 445, restrict the WebClient service, and limit outbound NTLM by policy. Upgrade SnakeYAML and restrict remote Java class loading.
3. **Inspect ICMP and odd TLS ports for tunnels.** Alert on QUIC headers inside ICMP Echo requests and on unapproved Ligolo-ng or TCP-to-UDP relay processes.

The Study in Numbers

Number	What it is
209	Endpoints across six tool families, on 175 distinct IPv4 hosts
23	Confirmed endpoints to block (11.0%): 16 JavaChains and 7 Responder, on 23 distinct addresses
42	JavaChains endpoints; 38 on port 8011
31 on 15	Responder endpoints on 15 hosts (2.07 endpoints per host), led by 7-port and 6-port mail hubs
3	Exposed staging workbenches: Packet-to-Shell, Termirror ICMP and RuoYi SnakeYAML
317.8 MB	Out-of-band exploit callback log staged on one workbench
94	ARL-Next (75) and X-Ray (19) scanner endpoints, all held for review, none blocked

How to use this report.

Security leaders can read Chapters 1, 7 and 10. SOC and detection engineers should start with Chapters 7 to 9. CTI analysts will find the infrastructure and workbench detail in Chapters 2 to 6.

02

CHAPTER 02

Four Roles, Six Tool Families

IN SHORT

The 209 endpoints split into four roles: Java exploit delivery, NTLM credential capture, covert tunnels and external reconnaissance. Each role uses its own tools and ports.

INSIDE THIS CHAPTER

The four roles 07

Counts by family 08

Where the endpoints sit 08

02 Four Roles, Six Tool Families

The four roles

Perimeters now strip inbound admin ports and inspect plain reverse shells. Operators respond with layered infrastructure. Each layer does one job.

Role	Tools	What it does
Reconnaissance	ARL-Next, X-Ray	Maps external assets, Java frameworks and web gateways
Initial access	JavaChains, RuoYi SnakeYAML builders, `p2s-pure`	Triggers out-of-band class loading or memory-shell injection in Java apps
Credential capture	Responder	Collects coerced Windows `NetNTLMv2` handshakes over TLS, SMB, RDP and mail ports
Covert transport	Termirror, Ligolo-ng, RedWarden	Keeps interactive access alive over ICMP or TLS, behind reverse-proxy guardrails

This four-role picture is an analytical model. Each family was observed on its own. No address hosts two families, and nothing links the families to one operator. The model rests on how each tool works and on files staged together on the three workbenches. Workbenches 1 and 3 appear in Chapter 3; Workbench 2 appears in Chapter 5.

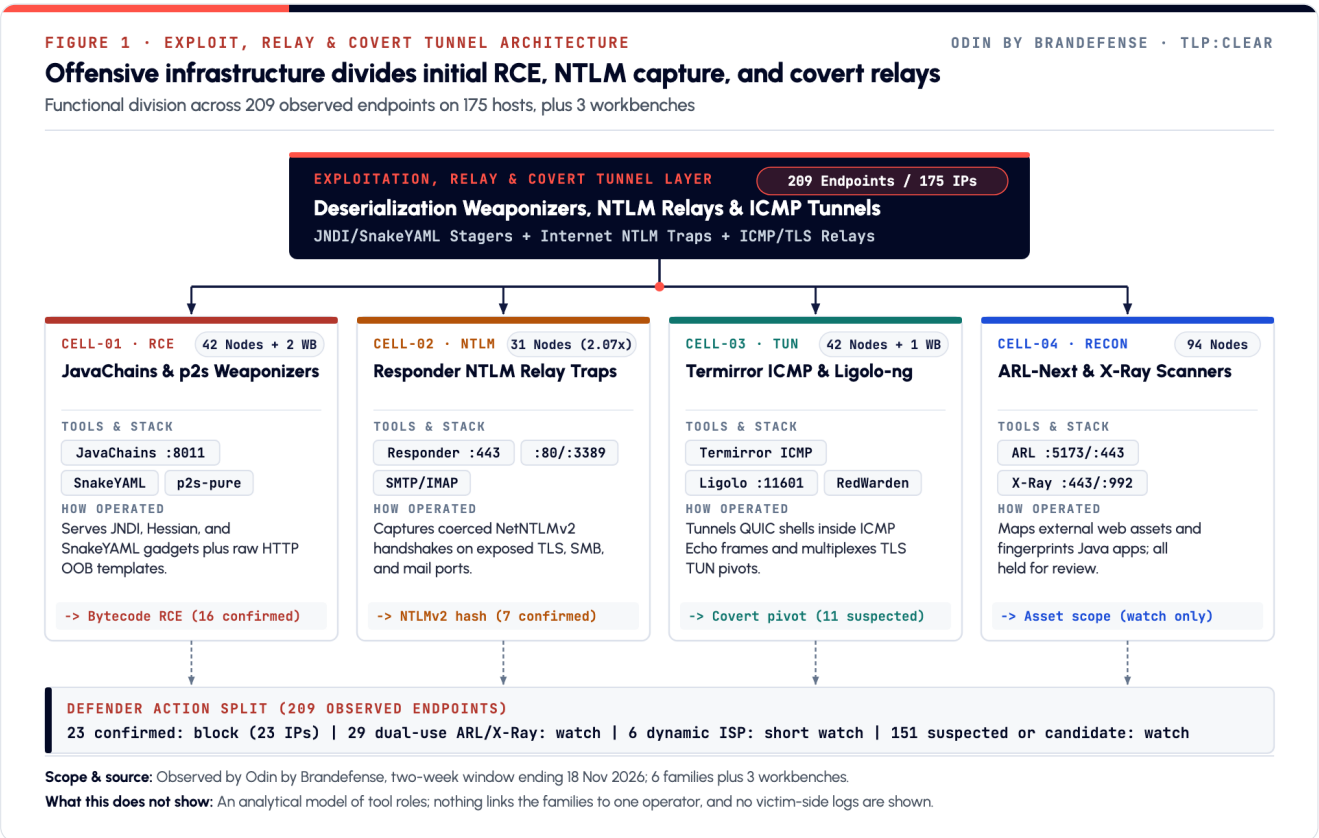


Figure 1: Offensive infrastructure divides initial RCE, NTLM capture, and covert relays

Counts by family

Each endpoint is an IP, port and tool combination.

Family	Role	Endpoints	Hosts	Endpoints per host	To block	Held for review
ARL-Next	Dual-use recon scanner	75	67	1.12	0	75 (15 dual-use confirmed, 2 dynamic ISP, 58 candidates)
JavaChains	Deserialization weaponizer	42	41	1.02	16	26 (3 dynamic ISP, 23 candidates)
Ligolo-ng	TLS tunnel relay	36	33	1.09	0	36 (7 suspected, 29 candidates)
Responder	NTLM capture	31	15	2.07	7	24 (23 suspected, 1 candidate)
X-Ray	Dual-use vulnerability scanner	19	13	1.46	0	19 (14 dual-use confirmed, 3 suspected, 2 candidates)
RedWarden	C2 reverse-proxy guardrail	6	6	1.00	0	6 (4 suspected, 1 dynamic ISP, 1 candidate)
Total		**209**	**175**		**23**	**186**

Twenty hosts expose more than one port. None hosts more than one family.

Where the endpoints sit

China hosts 87 endpoints (41.6%), the United States 46 (22.0%), Hong Kong 29 (13.9%) and Germany 15 (7.2%). India (7) and the Netherlands (6) follow.

Hosting follows tool lineage.

Family	Main location	Main hosting networks
JavaChains	China (36 of 42)	Shenzhen Tencent Computer Systems, Hangzhou Alibaba Advertising
ARL-Next	China and Hong Kong (63 of 75)	Shenzhen Tencent Computer Systems, Hangzhou Alibaba Advertising
Responder	United States (24 of 31)	Namecheap (14), DigitalOcean (7), Amazon (6)
Ligolo-ng	Spread	OVH SAS (6), Microsoft (3), DigitalOcean (2)

Across all families, Shenzhen Tencent hosts 48 endpoints and Hangzhou Alibaba 26.

CHAPTER 03

Java Deserialization Weaponizers

IN SHORT

A single unsafe JNDI, Hessian, JDBC or SnakeYAML parser lets an attacker load remote Java code. JavaChains servers and two exposed workbenches show how operators pair gadget servers with bulk out-of-band exploit templates.

INSIDE THIS CHAPTER

How the weaponizers work 10
The JavaChains fleet 10
Workbench 1: a Packet-to-Shell out-of-band workspace 11
Workbench 3: ARL reconnaissance next to a RuoYi SnakeYAML builder 12

03 Java Deserialization Weaponizers

How the weaponizers work

A simple proof-of-concept embeds one hard-coded command. An integrated weaponizer is different. It runs LDAP, RMI, HTTP and rogue MySQL or JDBC listeners at the same time. A vulnerable target then pulls a second-stage Java class over the wire.

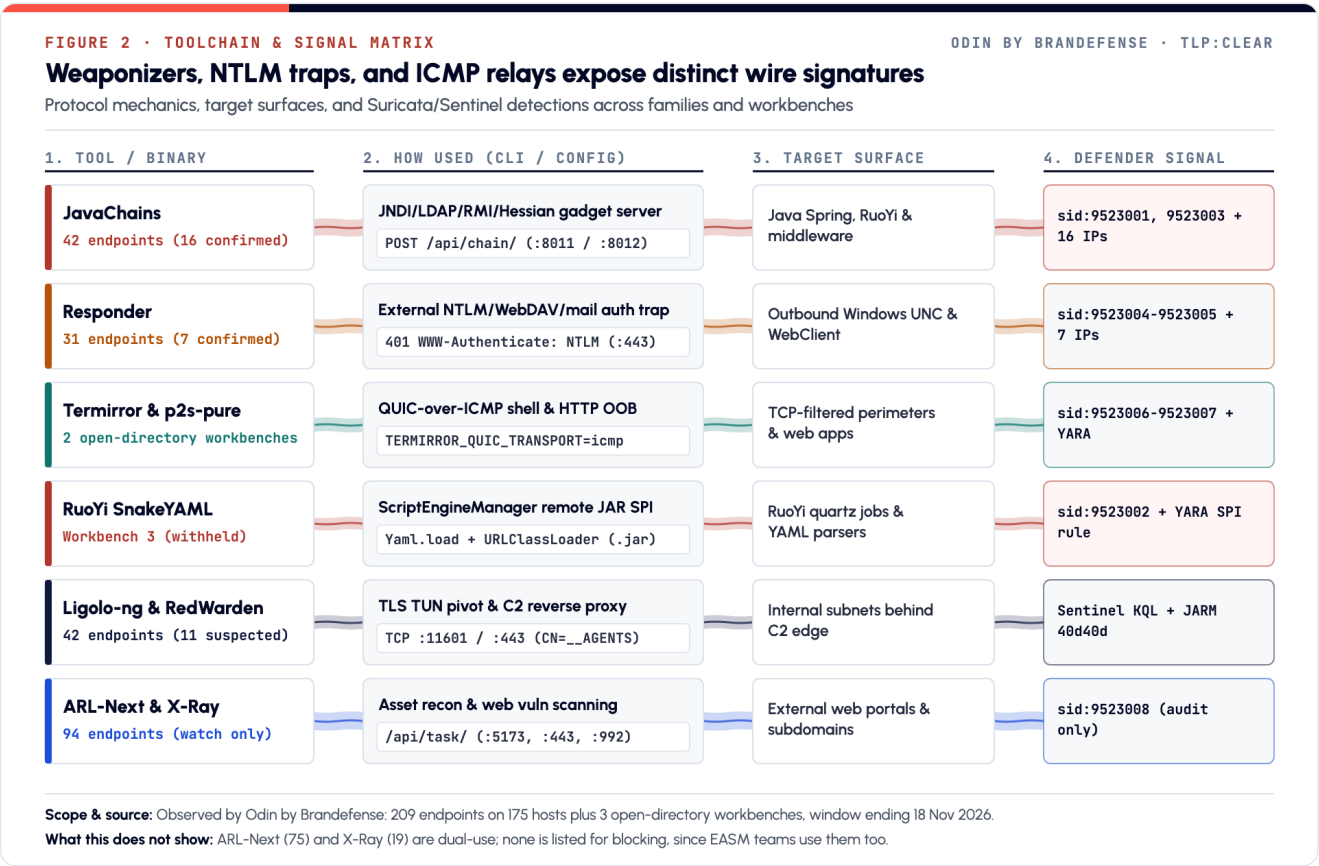


Figure 2: Weaponizers, NTLM traps, and ICMP relays expose distinct wire signatures

JavaChains bundles gadget generation for `CommonsCollections`, `CommonsBeanutils`, `Fastjson`, `Jackson`, `Hessian` and `SnakeYAML`. It adds built-in JNDI, LDAP and RMI callback listeners. A web control plane runs on TCP 8011 by default, with a secondary listener on 8012.

The JavaChains fleet

Of the 42 endpoints, 38 listen on 8011, 2 on 8012, and 1 each on 443 and 80.

The 16 confirmed servers run on dedicated cloud hosting.

Hosting network	Confirmed servers
Shenzhen Tencent Computer Systems (CN)	8
Hangzhou Alibaba Advertising (CN)	5
Volcano Engine (CN)	1
UCloud Information Technology (HK)	1
Hetzner Online (DE)	1

Fifteen of the 16 use 8011. The sixteenth, 49.233.77[.]89, fronts its weaponizer over TLS on 443. It shares JARM `2ad2ad0002ad2ad00042d42d00000000f78d2dc0ce6e5bbc5b8149a4872356` with a candidate host that listens on 8012 and 80. JARM is a fingerprint of how a server answers TLS handshakes.

Endpoint	Hosting	Country
124.220.216[.]115:8011	Shenzhen Tencent	CN
159.75.186[.]95:8011	Shenzhen Tencent	CN
128.1.131[.]142:8011	UCloud HK	HK
47.95.254[.]142:8011	Hangzhou Alibaba	CN
162.55.195[.]59:8011	Hetzner Online	DE
49.233.77[.]89:443	Shenzhen Tencent	CN

These are examples. All 16 appear in Chapter 9.

The feed marks three more JavaChains servers as confirmed, but they sit on China Unicom and China Mobile subscriber networks. This report holds them as dynamic-ISP endpoints, not confirmed C2. Their addresses rotate between customers, so they go on a 24-hour watchlist instead of a block list. Their addresses are not published.

Workbench 1: a Packet-to-Shell out-of-band workspace

Workbench 1 is an open directory on port 8888 of a host in China. Its address is withheld. It holds 123 files built around `p2s-pure`, an engine that replays raw HTTP exploit requests and records out-of-band callbacks.

File type	Count
`.bin` payloads	30
`.log` traces	20
`.resp` response captures	13
`.so` shared libraries	6
`.sh` runner scripts	5
`.req` request templates	4
`.txt` outputs	2

These counts cover 80 listed entries. The workspace has 14 randomized run folders named `/p2s-pure-<id>/` and `/p2s-remote-<id>/`. Each is built from raw request templates and stores a `leak.resp` response dump of 1.65 to 1.70 MB.

The operator does not craft each exploit by hand. Shell wrappers such as `remote_run.sh`, `remote_fetch_test.sh`, `remote_svc_check.sh`, `remote_rebuild8088.sh` and `remote_diag.sh` drive pre-built binary request blobs. Results are logged to `pwn_loop.log`, `pwn_single.log` and `oob.log`.

Staged file	Size (bytes)	Role
`/oob.log`	317,828,319	Out-of-band HTTP and DNS callback log (317.8 MB)
`/pkts/B1.req`	about 33.1 MB	Stage-B raw HTTP request template
`b1.bin`	33,128,677	Pre-built request blob
`/pkts/A1.req`	1,060,577	Stage-A raw HTTP request template
`a.bin`	1,060,211	Pre-built request blob
`/pkts/B1.resp`	1,668,549	Captured response dump
`/rev_lst.log`	255,094	Callback listener log for TCP 888
`/native_libs/libencrypt.so`	428,480	Staged crypto or JNI shared library

The six compiled ELF libraries include `liball_in_one.so`, `libapp.so`, `libencrypt.so`, `libcpatch.so` and `libAesJni.so`. The listener log on TCP 888 records 26 or more inbound reverse connections.

These are staged files and attacker logs. They show tooling and intent. We did not confirm any exploit against a specific target.

Workbench 3: ARL reconnaissance next to a RuoYi SnakeYAML builder

Workbench 3 is an open directory on a host in China. Its address is withheld. The directory lists 224 files: 126 `.jpg` screenshots of web login pages and 98 build, source and configuration files.

Two toolsets sit side by side. The first is `ARL-plus-docker`, a packaged reconnaissance worker with `wait-for-it.sh` and a self-signed certificate script, `gen_crt.sh`. The second is a RuoYi SnakeYAML weaponizer project, `yaml-payload-for-ruoyi`.

Item in the RuoYi project	Count
`.jar` archives	5
`ScriptEngineFactory` service files	7
`.java` source files	9
`.class` files	13

RuoYi is a popular open-source Java admin framework. Its scheduled-task feature can pass attacker text to `org.yaml.snakeyaml.Yaml.load`. The payload abuses this to create `javax.script.ScriptEngineManager` around `java[.]net.URLClassLoader`. The class loader then fetches a remote `.jar` declared under `META-INF/services/javax.script.ScriptEngineFactory`.

The project is designed to register an in-memory filter on routes such as `/login` or `/dev-api/`. No JSP file is written to disk. Suricata SID 9523002 looks for this pattern in HTTP request bodies.

The unsafe constructor behind this attack is tracked as CVE-2022-1471. The CVE record says SnakeYAML's `Constructor()` class does not restrict which types it creates during deserialization. It recommends `SafeConstructor` for untrusted input and an upgrade to SnakeYAML 2.0 or later. The CNA score is CVSS 3.1 8.3 (High).

CHAPTER 04

Internet-Exposed Responder NTLM Traps

IN SHORT

Responder is built for internal networks, where it answers name-resolution broadcasts. Here it runs on public cloud servers to collect Windows NTLM handshakes that internal hosts are tricked into sending out.

INSIDE THIS CHAPTER

How the trap works 15

The fleet 15

Seven confirmed listeners 15

Mail and FTP authentication hubs 16

04 Internet-Exposed Responder NTLM Traps

How the trap works

An internal Windows host can be made to open an attacker-supplied path. Examples are a UNC path, an Office template link, an SSRF redirect or a WebClient `DAVWWWRoot` URL. If that path points to an external address, Windows starts an NTLM challenge-response exchange on its own.

This only works when egress rules let the traffic out. The usual paths are SMB (445), HTTP or HTTPS (80 and 443), and some mail or RPC ports. The captured `NetNTLMv2` response can then be cracked offline or relayed.

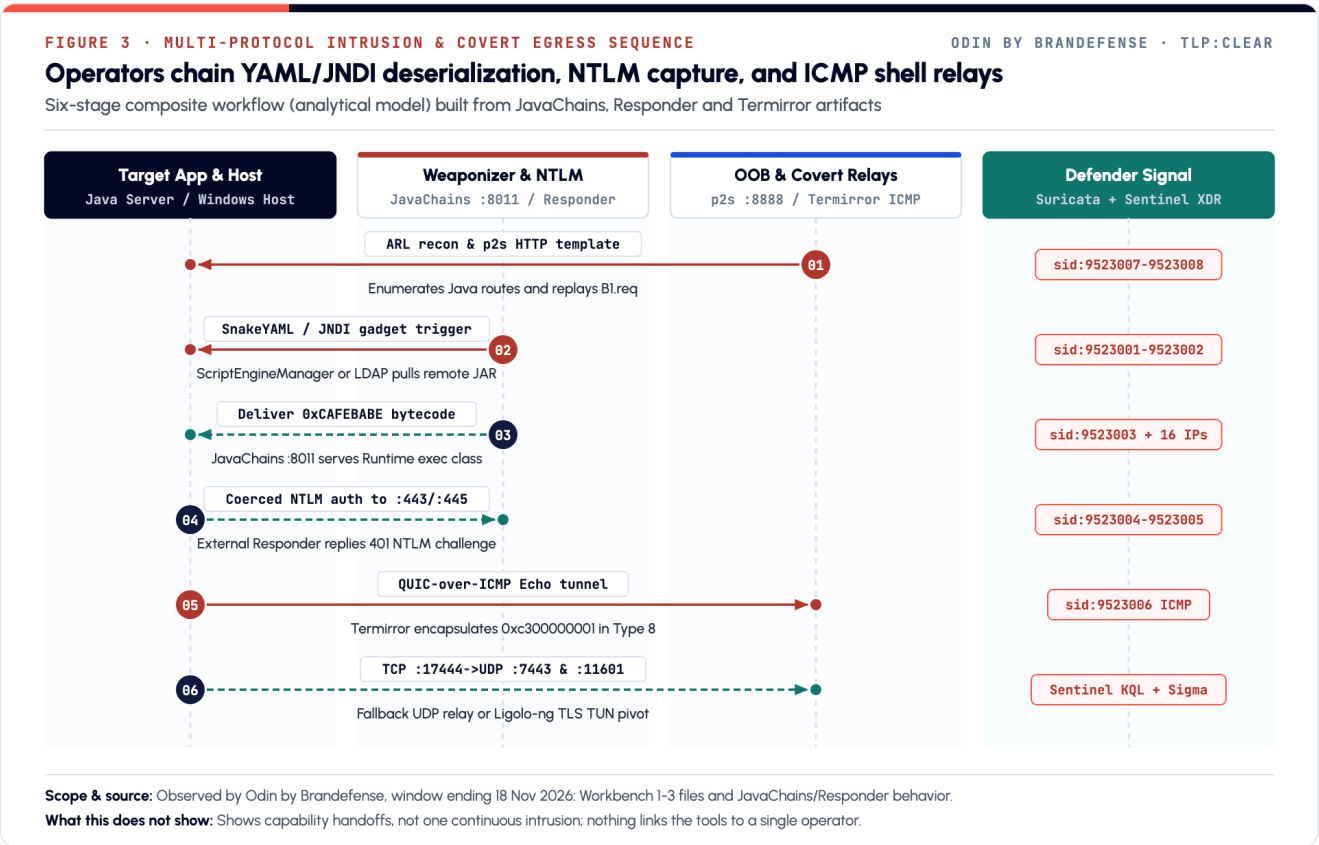


Figure 3: Operators chain YAML/JNDI deserialization, NTLM capture, and ICMP shell relays

Figure 3 shows how a deserialization foothold, an external NTLM trap and an ICMP tunnel could chain together. It is a composite of separate tool behaviors, not one observed intrusion.

The fleet

Responder has the highest port density in this study: 31 endpoints on 15 hosts, or 2.07 per host. Most sit in the United States (24) and Germany (4). Namecheap (14), DigitalOcean (7) and Amazon (6) host most of them.

Seven confirmed listeners

Four confirmed listeners use TCP 443. There, an `HTTP 401` reply with `WWW-Authenticate: NTLM` asks the client to authenticate. This gets through firewalls that block outbound SMB but allow HTTPS and WebDAV. Suricata SID 9523004 matches that reply.

The other three bind TLS sockets that can carry NTLM on non-web ports: remote desktop (3389), SIP over TLS (5061) and 5120.

Endpoint	Hosting	Country	Note
63.250.41[.]138:443	Namecheap	US	HTTPS NTLM challenge
142.93.190[.]214:443	DigitalOcean	US	HTTPS NTLM challenge
167.71.162[.]248:443	DigitalOcean	US	HTTPS NTLM challenge
167.172.40[.]16:443	DigitalOcean	NL	HTTPS NTLM challenge
193.149.176[.]161:3389	BL Networks	US	RDP listener, JARM `14d14d16d14d14d08c14d14d14d14dfd9c9d14e4f4f67f94f0359f8b28f532`
165.232.108[.]62:5061	DigitalOcean	GB	SIP over TLS listener
66.109.142[.]164:5120	Blackfoot Telephone	US	TLS listener

Mail and FTP authentication hubs

The 23 suspected Responder endpoints include two Namecheap servers in the United States. They share one TLS certificate and expose Responder's full rogue mail and FTP suite. Both addresses are withheld.

Hub	Ports
7-port hub	FTP 21, POP3 110, IMAP 143, SMTPS 465, SMTP submission 587, IMAPS 993, POP3S 995
6-port hub	21, 110, 143, 587, 993, 995
DigitalOcean mail listener	143, 465, 993
Three Amazon hosts	80 and 443 on each

A mail client that tries plain or NTLM login against these ports hands over credentials. Treat outbound connections from user subnets to unknown mail servers on these ports as suspicious.

05

CHAPTER 05

Covert ICMP Shells: the Termirror Workbench

IN SHORT

Many firewalls send web traffic through a proxy but still allow ping. Workbench 2 holds scripts that carry multiplexed QUIC shells and file transfers inside ICMP Echo packets, with a TCP-to-UDP fallback relay.

INSIDE THIS CHAPTER

What the workbench holds 18

What a defender can see 19

05 Covert ICMP Shells: the Termirror Workbench

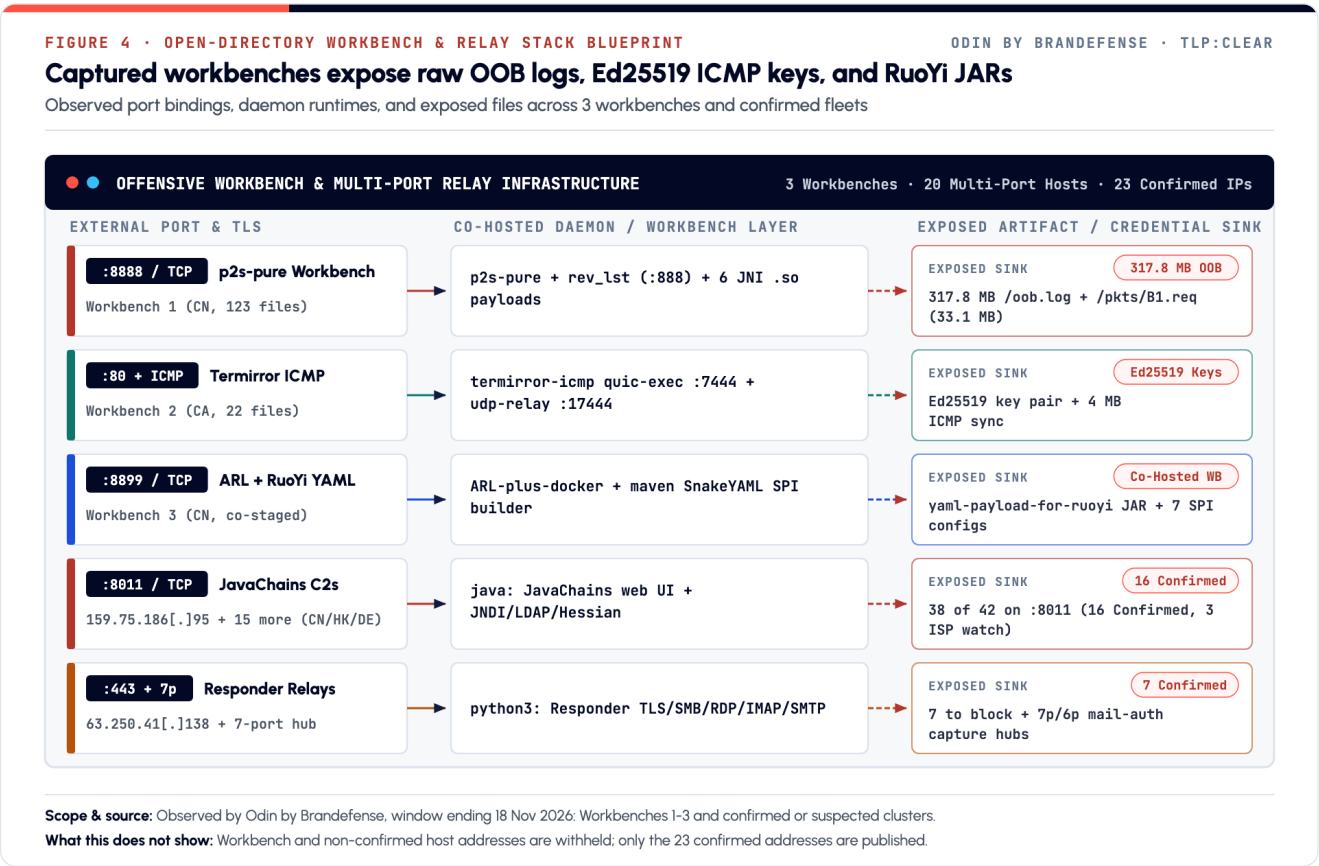


Figure 4: Captured workbenches expose raw OOB logs, Ed25519 ICMP keys, and RuoYi JARs

What the workbench holds

Workbench 2 is an open directory on port 80 of a host in Canada. Its address is withheld. It lists 42 entries across 30 distinct paths, with 22 unique operational files at the root.

The files form a complete covert-shell kit built around Termirror. The ATT&CK technique is Protocol Tunneling (T1572).

Mutual authentication keys. Two `.termirror` folders, one for a server and one for a client, each hold a 1,860-byte `identity.pub` public key, a 128-byte `identity.key` private key and a pinned `known_servers` list (92 and 473 bytes). An 81-byte `ssh-ed25519` recovery public key sits at the root.

QUIC over ICMP. Two scripts, `termirror-icmp-stress.sh` and `termirror-icmp-smoke.sh`, set `TERMIRROR_QUIC_TRANSPORT=icmp` and `TERMIRROR_QUIC_DIRECT=1`. They call `termirror-icmp-test quic-exec` against a remote peer on port 7444. The peer address is withheld.

The stress script is written to open 8 concurrent shell streams, with outputs `icmp-parallel-1.out` to `icmp-parallel-8.out`, and 10 sequential latency checks. It then pushes a 256 KB (262,144-byte) and a 4 MB (4,194,304-byte) binary file with `sync --to`. All of this travels inside ICMP Echo and Echo Reply frames.

TCP-to-UDP fallback. Middleboxes sometimes drop raw ICMP or UDP QUIC packets. For that case, `termirror-udp-relay-remote.py` (1,776 bytes) listens on TCP 17444. It reads 4-byte big-endian length headers, up to 65,535 bytes per frame, and forwards each datagram to local UDP 7443. A matching local client wrapper probably runs on the operator's side.

Staged file	Size (bytes)	Role
`identity.pub` (server and client)	1,860 each	Ed25519 identity public keys
`known_servers` (client)	473	Pinned server list
`termirror-icmp-stress.sh`	1,839	8-stream and 4 MB ICMP test
`termirror-udp-relay-remote.py`	1,776	TCP 17444 to UDP 7443 relay
`termirror-icmp-smoke.sh`	1,765	256 KB ICMP transfer check
`termirror-udp-relay-remote.cpython-312.pyc`	4,051	Python 3.12 bytecode of the relay
SSH recovery public key	81	`ssh-ed25519` key

The scripts show what the operator built and tested. We did not see this tunnel carry traffic out of any victim network.

What a defender can see

A QUIC version 1 long-header packet sets the two top bits of its first byte and carries the version bytes 00 00 00 01 right after it. Termirror places that header inside the data field of an ICMP Type 8 request. Ordinary ping payloads do not look like this. Suricata SID 9523006 alerts when one source sends 8 such packets within 30 seconds.

CHAPTER 06

Tunnels, Guardrails and Dual-Use Scanners

IN SHORT

Ligolo-ng and RedWarden provide encrypted pivots and C2 protection. ARL-Next and X-Ray map targets, but many security teams run them legitimately, so none of their 94 endpoints is listed for blocking.

INSIDE THIS CHAPTER

Ligolo-ng TLS tunnel relays 21

RedWarden C2 guardrails 21

ARL-Next and X-Ray 21

What not to block 22

06 Tunnels, Guardrails and Dual-Use Scanners

Ligolo-ng TLS tunnel relays

Ligolo-ng replaces slow SOCKS proxies with a userland network interface built on `gVisor`, tunneled over TLS. Odin found 36 endpoints on 33 hosts. Fourteen use 443 and ten use the tool's usual port 11601. One more host binds 11602 and 11603.

Two fingerprints help hunters.

Fingerprint	Where seen
JARM `40d40d40d00040d00043d43d00043d3bef2bf79cd6719851e8198c1e8f9a14`	Nine Ligolo-ng endpoints; a Go TLS stack fingerprint
Certificate `CN=__AGENTS`	Ligolo-ng's built-in `-selfcert` certificate, seen on a suspected relay on 4444 in Vietnam

The Go JARM value is shared by many Go programs, so use it only together with a second signal. Seven Ligolo-ng endpoints are suspected and 29 are candidates. None is confirmed, and their addresses are withheld.

RedWarden C2 guardrails

RedWarden is a reverse proxy that sits in front of a C2 team server. It checks HTTP headers, TLS fingerprints and source geography before it lets a request through. Odin found 6 endpoints.

Group	Endpoints	Detail
Suspected	4	OVH, DigitalOcean and Shock Hosting, on 81, 8080, 8083 and 8443
Dynamic telecom ISP	1	Chinanet, port 7777
Candidate	1	Port 443

ARL-Next and X-Ray

ARL-Next is an asset reconnaissance suite. Of its 75 endpoints, 45 use the Vite UI port 5173, 9 use 443, 6 use 80 and 4 use 8888. X-Ray is a web vulnerability scanner. Of its 19 endpoints, 8 use 443. Three Hong Kong hosts each expose 443, 992 and 5555 under one shared certificate.

Both tools map subdomains and web flaws. Both are also widely used by authorized attack-surface and bug-bounty teams. None of their hosts shares an address with a confirmed JavaChains or Responder listener. All 94 endpoints are therefore held for review, not blocked.

What not to block

The table gives an action for each handling group. Only the first group is ready to block.

Group	Endpoints	Action
Confirmed JavaChains and Responder on dedicated hosting	23	Block, review after 30 days
Confirmed dual-use scanners (ARL-Next 15, X-Ray 14)	29	Watch, do not block
Dynamic telecom addresses (3 JavaChains, 2 ARL-Next, 1 RedWarden)	6	Short watchlist of about 24 hours
Suspected (Responder 23, Ligolo-ng 7, RedWarden 4, X-Ray 3)	37	Watch, look for a second signal
Candidates	114	Hold until corroborated

Cloud addresses are reassigned over time. Re-check each blocked address before a block rule is renewed.

07

CHAPTER 07

What Defenders Should Do

IN SHORT

This chapter lists the actions defenders should take, in priority order.

07 What Defenders Should Do

Within 24 hours

- [ ] Block the 23 confirmed JavaChains and Responder addresses at perimeter firewalls and web gateways. Review them after 30 days.
- [ ] Put dynamic-ISP addresses on a short watchlist, not a block list.
- [ ] Keep ARL-Next and X-Ray signals in alert-only mode.

Within 72 hours

- [ ] Deny outbound SMB on 445 and 139 from all internal subnets.
- [ ] Restrict auto-start of the Windows WebClient (WebDAV) service, and limit outbound NTLM with the `RestrictSendingNTLMTraffic` policy.
- [ ] Rate-limit or block outbound ICMP Echo from server subnets that have no need for it.

Within 7 to 30 days

- [ ] Upgrade SnakeYAML to 2.0 or later and use `SafeConstructor` in RuoYi and Spring apps.
- [ ] Turn on JDK `ObjectInputFilter` serialization filters, and block outbound JNDI, LDAP and RMI callbacks from Java servers.
- [ ] Deploy the Suricata, KQL, Sigma and YARA rules in Chapter 8, with allowlists for named red-team and scanner hosts only.
- [ ] Add the JARM values and `CN=__AGENTS` to external attack-surface hunting, always paired with a second signal.

CHAPTER 08

Detection Rules

IN SHORT

This chapter describes each published rule and where to deploy it.

INSIDE THIS CHAPTER

How the Suricata signatures work 26

How the other rules work 26

08 Detection Rules

The full rules are in Appendix B. The only exclusions are exact hostnames in the KQL and Sigma allowlists.

Rule file	What it catches
`odin_javachains_responder_icmp_tunnels.rules`	Suricata 7: eight signatures for JavaChains, SnakeYAML, Responder NTLM, Termirror ICMP, `p2s-pure` and ARL-Next traffic
`sentinel_javachains_responder_ntlm_icmp_hunt.kql`	Microsoft Sentinel and Defender XDR: contact with the 23 confirmed addresses, relay ports, ICMP bursts and matching processes
`sigma_javachains_responder_termirror_execution.yml`	Sigma (3 rules): Linux processes for JavaChains and SnakeYAML builders, Termirror relays, and Responder, Ligolo-ng, ARL or X-Ray
`odin_javachains_p2s_termirror_ruoyi_artifacts.yar`	YARA (4 rules): JavaChains configs, `p2s-pure` workbench files, Termirror scripts and RuoYi SnakeYAML SPI builders

How the Suricata signatures work

SID	Looks for	Why
9523001	Outbound `POST` to `/api/chain/` on 8011, 8012 or 443 with `"gadget"` in the body	An internal host driving a JavaChains control API
9523002	Inbound HTTP body with `javax.script.ScriptEngineManager` followed by `java[.]net.URLClassLoader`	RuoYi or Spring SnakeYAML remote JAR load
9523003	HTTP 200 from 8011, 8012, 80 or 8899 whose body starts with `CA FE BA BE` and contains `java/lang/Runtime`	A Java class file with command execution delivered to an internal host
9523004	HTTP 401 from an external server with `WWW-Authenticate: NTLM`	An internet NTLM challenge trap
9523005	Outbound `NTLMSSP` negotiate message on 445, 139, 3389, 5061 or 5120	Windows authentication leaving the network
9523006	ICMP Type 8 with QUIC v1 Initial bytes `c3 00 00 00 01`, 8 times in 30 seconds from one source	Termirror QUIC-over-ICMP tunnel (T1572)
9523007	Outbound request with `/p2s-pure` in the URI	Contact with a Packet-to-Shell workbench
9523008	Outbound `/api/task/` with a `Token:` header on ARL ports	ARL-Next use; audit only, since the tool is dual-use

Exempt documented external gateways, such as Azure AD Application Proxy, from SIDs 9523004 and 9523005 by exact destination address. Keep SID 9523008 in alert-only mode.

How the other rules work

The KQL query looks back 7 days. It flags devices that contact any of the 23 confirmed addresses. It also flags use of relay ports such as 8011, 11601 and 17444, or 25 or more public ICMP events, on hosts not in the allowlist. It joins these with process events that name JavaChains, Termirror, `p2s-pure`, Ligolo-ng or `Responder.py`.

The allowlist holds exact hostnames only. It never hides a match on a confirmed address. Each row gets an action: block, isolate a covert tunnel, or hunt a callback or NTLM leak. Network monitoring appliances that send many pings can be added to the allowlist by exact name.

The three Sigma rules need Linux process-creation logs. Rule 1 flags `java`, `mvn`, `python3` or a shell whose command line names `javachains`, `yaml-payload-for-ruoyi` or `javax.script.ScriptEngineManager`. Rule 2 flags Termirror ICMP and relay command lines. Rule 3 flags `Responder.py`, Ligolo-ng on 11601 or with `-selfcert`, `ARL-plus-docker` or `xray_linux_amd64`, at medium level.

Each YARA rule is limited to files under 512 KB and needs strings from at least two groups.

YARA rule	Fires when
`ODIN_JavaChains_Deserialization_Controller_Config`	A JavaChains name, a 8011 or 8012 port string, and `Hessian` or `jndi` appear together
`ODIN_P2S_Pure_PacketToShell_Workbench_Manifest`	A `p2s-pure` or `p2s-remote` name, a listener or OOB log name, and `libAesJni.so` or `leak.resp` appear together
`ODIN_Termirror_Covert_ICMP_QUIC_Relay_Script`	`quic-exec` appears with a Termirror ICMP flag, or with both relay ports
`ODIN_RuoYi_SnakeYAML_SPI_Gadget_Manifest`	A `ScriptEngineManager` or SPI service path appears with `URLClassLoader`, the SnakeYAML class or the RuoYi payload name

09

CHAPTER 09

Indicators

IN SHORT

This chapter lists the published indicators and explains what stays withheld.

INSIDE THIS CHAPTER

Confirmed endpoints to block 29

09 Indicators

Appendix A holds 23 published indicators: 23 addresses and no hashes. These are the 16 confirmed JavaChains and 7 confirmed Responder endpoints.

Type	Count	Where
Confirmed endpoints to block (IP and port)	23	Appendix A and the table below
Workbench and multi-port watchlist hosts	5	Withheld
File hashes (SHA-256)	0	All workbench file hashes are withheld

Supporting fingerprints stay in the text only: the JARM values in Chapters 3, 4 and 6, the certificate `CN=__AGENTS`, and the QUIC-in-ICMP byte pattern.

Withheld: the three workbench addresses, the mail hubs, the Ligolo-ng relay presenting `CN=__AGENTS`, all dynamic-ISP addresses, all suspected and candidate addresses, the tool certificate domains, and the hashes of all workbench files. The workbench owners are not established.

Confirmed endpoints to block

Endpoint	Family	Hosting	Country
124.220.216[.]115:8011	JavaChains	Shenzhen Tencent	CN
159.75.186[.]95:8011	JavaChains	Shenzhen Tencent	CN
111.229.216[.]164:8011	JavaChains	Shenzhen Tencent	CN
128.1.131[.]142:8011	JavaChains	UCloud HK	HK
47.95.254[.]142:8011	JavaChains	Hangzhou Alibaba	CN
8.134.112[.]14:8011	JavaChains	Hangzhou Alibaba	CN
124.223.36[.]16:8011	JavaChains	Shenzhen Tencent	CN
162.55.195[.]59:8011	JavaChains	Hetzner Online	DE
122.51.118[.]220:8011	JavaChains	Shenzhen Tencent	CN
39.106.171[.]189:8011	JavaChains	Hangzhou Alibaba	CN
49.233.77[.]89:443	JavaChains	Shenzhen Tencent	CN
101.96.216[.]248:8011	JavaChains	Volcano Engine	CN
124.221.179[.]152:8011	JavaChains	Shenzhen Tencent	CN
120.76.158[.]8:8011	JavaChains	Hangzhou Alibaba	CN
47.103.214[.]7:8011	JavaChains	Hangzhou Alibaba	CN
106.54.13[.]23:8011	JavaChains	Shenzhen Tencent	CN
63.250.41[.]138:443	Responder	Namecheap	US

Endpoint	Family	Hosting	Country
142.93.190[.]214:443	Responder	DigitalOcean	US
66.109.142[.]164:5120	Responder	Blackfoot Telephone	US
193.149.176[.]161:3389	Responder	BL Networks	US
165.232.108[.]62:5061	Responder	DigitalOcean	GB
167.71.162[.]248:443	Responder	DigitalOcean	US
167.172.40[.]16:443	Responder	DigitalOcean	NL

10

CHAPTER 10

Scope and Limits

IN SHORT

This chapter sets out what the evidence can show and what it cannot.

10 Scope and Limits

- **Window.** The findings cover a two-week observation window ending 18 November 2026. Some records carry older first-seen dates from public sources. We do not use those as a tracking start.
- **The tool, not the victim.** A confirmed endpoint runs JavaChains or Responder. We did not see these servers succeed against any named target.
- **The four roles are a model.** No address hosts two families, and nothing links the families or workbenches to one operator.
- **Workbenches show intent.** Their files are staged tools, templates and attacker logs. Their owners are not established, so their addresses and file hashes are withheld.
- **Dual-use tools are not proof of attack.** ARL-Next, X-Ray, Ligolo-ng and RedWarden all have legitimate users. Treat them as leads.
- **Rule testing.** Every published rule file loads without errors in its engine (Suricata 7.0.12, YARA-X, pySigma and Microsoft's Kusto.Language analyzer, checked on 20 November 2026). The rules were not replayed against live traffic, so test and tune them in your own environment before production use.

11

CHAPTER 11

Sources

IN SHORT

This chapter lists the public references used in the report.

11 Sources

1. OWASP Foundation. Deserialization Cheat Sheet. https://cheatsheetseries.owasp.org/cheatsheets/Deserialization_Cheat_Sheet.html (verified 18 November 2026)
2. NIST National Vulnerability Database. CVE-2022-1471 Detail. https://nvd.nist.gov/vuln/detail/CVE-2022-1471 (verified 18 November 2026)
3. CVE Program. CVE-2022-1471 record. https://www.cve.org/CVERecord?id=CVE-2022-1471 (verified 20 November 2026)
4. Microsoft Learn. NTLM overview. https://learn.microsoft.com/en-us/windows-server/security/kerberos/ntlm-overview (verified 18 November 2026)
5. MITRE ATT&CK. Protocol Tunneling, T1572. https://attack.mitre.org/techniques/T1572/ (verified 18 November 2026)
6. OISF. Suricata 7.0 User Guide, HTTP keywords. https://docs.suricata.io/en/suricata-7.0.0/rules/http-keywords.html (verified 18 November 2026)
7. Salesforce Engineering. Easily Identify Malicious Servers on the Internet with JARM. https://engineering.salesforce.com/easily-identify-malicious-servers-on-the-internet-with-jarm-e095edac525a/ (verified 18 November 2026)

APPENDICES

Indicators and Detection Rules

FOR DEFENDERS

Appendix A lists the indicators published for this study. Appendix B carries every published detection rule in full.

INSIDE

A. Indicators of compromise (23) 36

B. Detection rules (4 files) 37

A Indicators of Compromise

Values are defanged for display. The CSV, JSON and STIX files on the online edition carry the raw values. Values the report marks as withheld are not listed.

ID	Type	Indicator	Port	Context	Action
R23-IOC-01	IPv4	124.220.216[.]115	8011	javachains	BLOCK
R23-IOC-02	IPv4	159.75.186[.]95	8011	javachains	BLOCK
R23-IOC-03	IPv4	111.229.216[.]164	8011	javachains	BLOCK
R23-IOC-04	IPv4	128.1.131[.]142	8011	javachains	BLOCK
R23-IOC-05	IPv4	47.95.254[.]142	8011	javachains	BLOCK
R23-IOC-06	IPv4	8.134.112[.]14	8011	javachains	BLOCK
R23-IOC-07	IPv4	124.223.36[.]16	8011	javachains	BLOCK
R23-IOC-08	IPv4	162.55.195[.]59	8011	javachains	BLOCK
R23-IOC-09	IPv4	122.51.118[.]220	8011	javachains	BLOCK
R23-IOC-10	IPv4	39.106.171[.]189	8011	javachains	BLOCK
R23-IOC-11	IPv4	49.233.77[.]89	443	javachains	BLOCK
R23-IOC-12	IPv4	101.96.216[.]248	8011	javachains	BLOCK
R23-IOC-13	IPv4	124.221.179[.]152	8011	javachains	BLOCK
R23-IOC-14	IPv4	120.76.158[.]8	8011	javachains	BLOCK
R23-IOC-15	IPv4	47.103.214[.]7	8011	javachains	BLOCK
R23-IOC-16	IPv4	106.54.13[.]23	8011	javachains	BLOCK
R23-IOC-17	IPv4	63.250.41[.]138	443	responder	BLOCK
R23-IOC-18	IPv4	142.93.190[.]214	443	responder	BLOCK
R23-IOC-19	IPv4	66.109.142[.]164	5120	responder	BLOCK
R23-IOC-20	IPv4	193.149.176[.]161	3389	responder	BLOCK
R23-IOC-21	IPv4	165.232.108[.]62	5061	responder	BLOCK
R23-IOC-22	IPv4	167.71.162[.]248	443	responder	BLOCK
R23-IOC-23	IPv4	167.172.40[.]16	443	responder	BLOCK

B Detection Rules

Each rule file loads in its own engine (YARA-X, Suricata 7, Kusto, pySigma). Tune thresholds and allow-lists to your environment before alerting or blocking. The same files can be downloaded from the online edition.

odin_javachains_p2s_termirror_ruoyi_artifacts.yar

YARA

```
/*
   Odin by Brandefense - Monograph 23 / 30 YARA Artifact Suite
   Scope: Configuration, relay, and workbench artifacts for JavaChains,
          p2s-pure Packet-to-Shell, Termirror QUIC-over-ICMP, and RuoYi SnakeYAML SPI.
   Classification: TLP:CLEAR
*/

rule ODIN_JavaChains_Deserialization_Controller_Config
{
    meta:
        author = "Odin by Brandefense"
        monograph = "23 / 30"
        classification = "TLP:CLEAR"
        description = "Matches JavaChains JNDI/LDAP/RMI/Hessian/JDBC weaponizer configuration manifests"
    strings:
        $jc1 = "javachains" ascii nocase
        $jc2 = "JavaChains" ascii
        $pt1 = "8011" ascii
        $pt2 = "8012" ascii
        $gd1 = "Hessian" ascii nocase
        $gd2 = "jndi" ascii nocase
    condition:
        filesize < 512KB and ($jc1 or $jc2) and ($pt1 or $pt2) and ($gd1 or $gd2)
}

rule ODIN_P2S_Pure_PacketToShell_Workbench_Manifest
{
    meta:
        author = "Odin by Brandefense"
        monograph = "23 / 30"
        classification = "TLP:CLEAR"
        description = "Matches p2s-pure Packet-to-Shell orchestration scripts and reverse listener logs"
    strings:
        $p1 = "p2s-pure" ascii
        $p2 = "p2s-remote" ascii
        $l1 = "rev_lst.log" ascii
        $l2 = "oob.log" ascii
        $n1 = "libAesJni.so" ascii
        $n2 = "leak.resp" ascii
    condition:
        filesize < 512KB and ($p1 or $p2) and ($l1 or $l2) and ($n1 or $n2)
}

rule ODIN_Termirror_Covert_ICMP_QUIC_Relay_Script
{
    meta:
        author = "Odin by Brandefense"
        monograph = "23 / 30"
        classification = "TLP:CLEAR"
        description = "Matches Termirror QUIC-over-ICMP automation scripts and TCP-to-UDP relay wrappers"
    strings:
        $t1 = "TERMIRROR_QUIC_TRANSPORT=icmp" ascii
        $t2 = "TERMIRROR_QUIC_DIRECT=1" ascii
        $q1 = "quic-exec" ascii
        $u1 = "17444" ascii
        $u2 = "7443" ascii
    condition:
        filesize < 512KB and (($t1 and $q1) or ($t2 and $q1) or ($u1 and $u2 and $q1))
}
```

```
rule ODIN_RuoYi_SnakeYAML_SPI_Gadget_Manifest
{
    meta:
        author = "Odin by Brandefense"
        monograph = "23 / 30"
        classification = "TLP:CLEAR"
        description = "Matches RuoYi and Spring SnakeYAML ScriptEngineManager SPI gadget builder artifacts"
    strings:
        $s1 = "javax.script.ScriptEngineManager" ascii
        $s2 = "META-INF/services/javax.script.ScriptEngineFactory" ascii
        $u1 = "java.net.URLClassLoader" ascii
        $r1 = "org.yaml.snakeyaml.Yaml" ascii
        $r2 = "yaml-payload-for-ruoyi" ascii nocase
    condition:
        filesize < 512KB and ($s1 or $s2) and ($u1 or $r1 or $r2)
}
```

odin_javachains_responder_icmp_tunnels.rules

SURICATA

```
# Odin by Brandefense - Monograph 23 / 30 Suricata 7.x Detection Suite
# Coverage: JavaChains (:8011/:8012), RuoYi SnakeYAML SPI, Responder NTLM Traps,
#           Termirror QUIC-over-ICMP, p2s-pure OOB Staging, and ARL-Next (audit only)
# Classification: TLP:CLEAR

alert http $HOME_NET any -> $EXTERNAL_NET [8011,8012,443] (msg:"ODIN [R23] JavaChains Deserialization Weaponizer
Control API Dispatch"; flow:established,to_server; http.method; content:"POST"; http.uri; bsize:>11; content:"/
api/chain/"; fast_pattern; http.request_body; bsize:>20; content:"\"gadget\""; classtype:trojan-activity; sid:
9523001; rev:1; metadata:created_at 2026_11_18, deployment perimeter, tactic initial_access;)

alert http any any -> $HOME_NET any (msg:"ODIN [R23] RuoYi and Spring SnakeYAML ScriptEngineManager Remote JAR
Load"; flow:established,to_server; http.request_body; bsize:>48; content:"javax.script.ScriptEngineManager";
fast_pattern; content:"java.net.URLClassLoader"; distance:0; classtype:web-application-attack; sid:9523002; rev:
1; metadata:created_at 2026_11_18, deployment perimeter, tactic initial_access;)

alert http $EXTERNAL_NET [8011,8012,80,8899] -> $HOME_NET any (msg:"ODIN [R23] JavaChains Remote JNDI Class
Bytecode Stager Delivery"; flow:established,from_server; http.stat_code; content:"200"; http.response_body;
bsize:>32; content:"|ca fe ba be|"; startswith; content:"java/lang/Runtime"; fast_pattern; classtype:trojan-
activity; sid:9523003; rev:1; metadata:created_at 2026_11_18, deployment perimeter, tactic execution;)

alert http $EXTERNAL_NET [80,443,3128,8443] -> $HOME_NET any (msg:"ODIN [R23] Internet-Exposed Responder HTTP or
WebDAV NTLM Challenge Trap"; flow:established,from_server; http.stat_code; content:"401"; http.header; bsize:>22;
content:"WWW-Authenticate: NTLM"; fast_pattern; classtype:credential-theft; sid:9523004; rev:1;
metadata:created_at 2026_11_18, deployment perimeter, tactic credential_access;)

alert tcp $HOME_NET any -> $EXTERNAL_NET [445,139,3389,5061,5120] (msg:"ODIN [R23] Outbound NTLMSSP
Authentication Frame to External Internet Host"; flow:established,to_server; content:"NTLMSSP|00 01 00 00 00|";
fast_pattern; classtype:credential-theft; sid:9523005; rev:1; metadata:created_at 2026_11_18, deployment
perimeter, tactic credential_access;)

alert icmp $HOME_NET any -> $EXTERNAL_NET any (msg:"ODIN [R23] Termirror Covert QUIC-over-ICMP Echo Tunnel
Initial Frame"; itype:8; icode:0; byte_test:1,>=,0xC0,0; content:"|00 00 00 01|"; offset:1; depth:4;
fast_pattern; threshold:type both, track by_src, count 8, seconds 30; classtype:trojan-activity; sid:9523006;
rev:2; metadata:created_at 2026_11_18, deployment perimeter, tactic command_and_control;)

alert http $HOME_NET any -> $EXTERNAL_NET [80,8888,8088] (msg:"ODIN [R23] p2s-pure Packet-to-Shell Out-of-Band
Workbench Artifact Fetch"; flow:established,to_server; http.uri; bsize:>9; content:"/p2s-pure"; fast_pattern;
classtype:trojan-activity; sid:9523007; rev:1; metadata:created_at 2026_11_18, deployment perimeter, tactic
command_and_control;)

alert http $HOME_NET any -> $EXTERNAL_NET [80,443,5173,8888,8899] (msg:"ODIN [R23] Dual-Use ARL-Next and ARL-Plus
Asset Recon API (Audit Only)"; flow:established,to_server; http.uri; bsize:>10; content:"/api/task/";
fast_pattern; http.header; bsize:>8; content:"Token:"; classtype:policy-violation; sid:9523008; rev:1;
metadata:created_at 2026_11_18, deployment perimeter, policy dual_use_audit;)
```

sentinel_javachains_responder_ntlm_icmp_hunt.kql

KQL (SENTINEL / DEFENDER XDR)

```
// Odin by Brandefense - Monograph 23 / 30 Microsoft Sentinel & Defender XDR Hunt
// Correlates egress to confirmed JavaChains/Responder servers, external NTLM leaks, and covert tunnel relays
// Classification: TLP:CLEAR
// The host allowlist applies only to the port, ICMP and process branches; it never hides a confirmed IP match.

let Lookback = 7d;
// Placeholder host names: replace with your own approved hosts.
let ApprovedSecurityHosts = dynamic(["REDTEAM-RELAY-01", "EASM-SCANNER-01", "APPSEC-LAB-01"]);
let ConfirmedJavaChainsAndResponderIPs = dynamic([
    "159.75.186.95", "122.51.118.220", "49.233.77.89", "101.96.216.248",
    "124.221.179.152", "124.220.216.115", "111.229.216.164", "128.1.131.142",
    "47.95.254.142", "8.134.112.14", "124.223.36.16", "162.55.195.59",
    "39.106.171.189", "120.76.158.8", "47.103.214.7", "106.54.13.23",
    "63.250.41.138", "142.93.190.214", "66.109.142.164", "193.149.176.161",
    "165.232.108.62", "167.71.162.248", "167.172.40.16"
]);
let WeaponizerAndRelayPorts = dynamic([445, 1099, 1389, 3128, 5061, 5120, 7443, 7444, 8011, 8012, 11601, 17444]);
let UnapprovedTunnelAndChainProcs = DeviceProcessEvents
    | where Timestamp >= ago(Lookback)
    | where not(DeviceName in~ (ApprovedSecurityHosts))
    | where ProcessCommandLine has_any ("javachains", "ScriptEngineManager", "termirror",
"TERMIRROR_QUIC_TRANSPORT", "p2s-pure", "ligolo", "Responder.py")
        or (FileName in~ ("java", "python", "python3", "sh", "bash") and ProcessCommandLine has_any
("URLClassLoader", "quic-exec", "17444", "11601", "8011"))
    | project ProcTime = Timestamp, DeviceId, DeviceName, ProcFileName = FileName, ProcessCommandLine;
DeviceNetworkEvents
| where Timestamp >= ago(Lookback)
| where RemoteIPType == "Public"
| extend ConfirmedHit = RemoteIP in (ConfirmedJavaChainsAndResponderIPs)
| where ConfirmedHit
    or (not(DeviceName in~ (ApprovedSecurityHosts)) and (RemotePort in (WeaponizerAndRelayPorts) or Protocol =~
"Icmp"))
| summarize
    NetFirstSeen = min(Timestamp),
    NetLastSeen = max(Timestamp),
    DistinctRemoteIPs = dcount(RemoteIP),
    DistinctPorts = make_set(RemotePort, 16),
    IcmpEvents = countif(Protocol =~ "Icmp"),
    MatchedConfirmedIPs = make_set_if(RemoteIP, ConfirmedHit, 24),
    ConnectionCount = count()
    by DeviceId, DeviceName, InitiatingProcessFileName
| join kind=leftouter (UnapprovedTunnelAndChainProcs) on DeviceId, DeviceName
| where array_length(MatchedConfirmedIPs) > 0 or isnotempty(ProcessCommandLine) or IcmpEvents >= 25
| extend AlertAction = case(
    array_length(MatchedConfirmedIPs) > 0, "BLOCK_CONFIRMED_JAVACHAINS_OR_RESPONDER",
    IcmpEvents >= 25 or ProcessCommandLine has "termirror", "ISOLATE_COVERT_ICMP_OR_UDP_RELAY_TUNNEL",
    "HUNT_DESERIALIZATION_CALLBACK_OR_NTLM_LEAK"
)
| project NetFirstSeen, NetLastSeen, DeviceName, AlertAction, MatchedConfirmedIPs, IcmpEvents, DistinctRemoteIPs,
DistinctPorts, InitiatingProcessFileName, ProcFileName, ProcessCommandLine, ConnectionCount
| sort by ConnectionCount desc
```

sigma_javachains_responder_termirror_execution.yml

SIGMA

```
title: Unapproved JavaChains or SnakeYAML Deserialization Weaponizer Execution
id: 4b8e21c9-6d14-4a92-8f30-202611182301
status: experimental
description: Detects unapproved execution of JavaChains deserialization weaponizers (:8011/:8012) or RuoYi
SnakeYAML ScriptEngineManager remote JAR payload builders.
author: Odin by Brandefense
date: 2026/11/18
tags:
  - attack.initial_access
  - attack.execution
```

```
  logsource:
    category: process_creation
    product: linux
  detection:
    selection_runtime:
      Image|endswith:
        - '/java'
        - '/mvn'
        - '/python3'
        - '/sh'
        - '/bash'
    selection_cli:
      CommandLine|contains:
        - 'javachains'
        - 'yaml-payload-for-ruoyi'
        - 'javax.script.ScriptEngineManager'
    filter_approved_lab:
      # Placeholder host names: replace with your own approved hosts.
      ComputerName:
        - 'APPSEC-LAB-01'
        - 'REDTEAM-RELAY-01'
    condition: selection_runtime and selection_cli and not filter_approved_lab
  falsepositives:
    - Authorized application security deserialization testing sandboxes listed in asset inventory
  level: high
  ---
  title: Termirror Covert QUIC-over-ICMP Shell and TCP-to-UDP Relay Execution
  id: 8c3f59a4-1e72-4d08-9b65-202611182302
  status: experimental
  description: Identifies Termirror covert interactive shells operating over raw ICMP Echo/Reply frames
  (TERMIRROR_QUIC_TRANSPORT=icmp) or length-prefixed TCP-to-UDP relay wrappers (:17444 to :7443).
  author: Odin by Brandefense
  date: 2026/11/18
  tags:
    - attack.command_and_control
    - attack.t1572
  logsource:
    category: process_creation
    product: linux
  detection:
    selection_termirror:
      CommandLine|contains:
        - 'TERMIRROR_QUIC_TRANSPORT=icmp'
        - 'termirror-icmp'
        - 'quic-exec'
        - 'termirror-udp-relay'
    filter_approved_relay:
      # Placeholder host names: replace with your own approved hosts.
      ComputerName:
        - 'REDTEAM-RELAY-01'
    condition: selection_termirror and not filter_approved_relay
  falsepositives:
    - Sanctioned red-team covert egress validation exercises documented with security operations
  level: high
  ---
  title: Responder Poisoner Ligolo-ng Tunnel or Dual-Use ARL-Next Scanner Spawn
  id: 2e7a90d6-5c41-48f3-a612-202611182303
  status: experimental
  description: Detects process execution of Responder NTLM relay listeners, Ligolo-ng TLS pivot relays (:11601 / -
  selfcert), or unapproved ARL-Next and X-Ray vulnerability scanners.
  author: Odin by Brandefense
  date: 2026/11/18
  tags:
    - attack.credential_access
    - attack.command_and_control
    - attack.reconnaissance
  logsource:
    category: process_creation
    product: linux
  detection:
    selection_tools:
      CommandLine|re: '(Responder\.py|ligolo[-_](proxy|agent).*11601|ligolo.*-selfcert|ARL-plus-docker|
```

```
xray_linux_amd64)'
  filter_approved_scanners:
    # Placeholder host names: replace with your own approved hosts.
    ComputerName:
      - 'REDTEAM-RELAY-01'
      - 'EASM-SCANNER-01'
  condition: selection_tools and not filter_approved_scanners
falsepositives:
  - Authorized external attack surface management (ARL-Next / X-Ray) or red-team relay hosts; apply dual-use
watchlist review for scanner hits
level: medium
```

ODIN

About Odin Research

Odin Research is the public research series of Odin by Brandefense. It documents adversary infrastructure seen on the internet and turns each finding into indicators and detection rules that defenders can use. Victims and third parties stay anonymous, addresses appear only for confirmed adversary infrastructure, and no working attack code is published.

Odin is the agentic analyst inside the Brandefense threat intelligence platform, which unifies Cyber Threat Intelligence, Digital Risk Protection and External Attack Surface Management.

Online edition, indicators and rules · brandefense.io/odin-research/reports/java-weaponizers-and-ntlm-traps/
Short Read · brandefense.io/odin-research/blog/java-exploit-servers-internet-ntlm-traps-and-tunnels-hidden/
Defense Kit · brandefense.io/odin-research/defense-kit/