ODIN

TLP:CLEAR

ODIN RESEARCH · MONOGRAPH 25 / 30 · C2 INFRASTRUCTURE

FortiGate Edge Kits and Role-Named Implants

Staging From Perimeter to Core

How open staging servers pair a FortiGate 100E kit with multi-architecture gopher and beacon builds, next to modular C2 fleets

Two-week observation window ending 18 November 2026 · Data cutoff 19 November 2026 · Prepared November 2026
Full Report · Public edition · Version 1.1

71
Modular C2 and relay endpoints across ten families, on 68 hosts

8
Confirmed controllers: 6 Nemo, 1 Xalgorix, 1 AtlasX (7 addresses published)

3
Open staging servers on port 8000 in the US, Finland and China, holding 133 files

77
Files on Workbench 1, including the seven-file / fg100e/ kit

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

How open staging servers pair a FortiGate 100E kit with multi-architecture gopher and beacon builds, next to modular C2 fleets

01	Executive Summary	04
02	From the Edge to the Core	07
03	Workbench 1: The FortiGate 100E Kit and the Build Matrix	10
04	Workbenches 2 and 3: Sliver and SSRF Callbacks	15
05	Modular C2 Fleets: Nemo, Xalgorix, AtlasX and Relays	18
06	Who Runs This: Competing Explanations	22
07	What Defenders Should Do	24
08	Detection Rules	26
09	Indicators	29
10	Scope and Limits	31
11	Sources	33
A	Indicators of compromise (7)	35
B	Detection rules (4 files)	36

IN SHORT

In a two-week observation window ending 18 November 2026, with data to 19 November, Odin by Brandefense found three open staging servers on port 8000. The largest holds a seven-file kit aimed at FortiGate 100E firewalls, 26 Adaptix gopher builds and 22 Windows beacons. Some builds are named for DRBD storage and Tomcat servers. Separately, 71 modular C2 endpoints sit on 68 hosts; 8 are confirmed. Block the 7 published controllers, check firewall integrity offline, and watch Linux storage and Tomcat hosts.

01

CHAPTER 01

Executive Summary

IN SHORT

In a two-week observation window ending 18 November 2026, with data to 19 November, Odin by Brandefense found three open staging servers on port 8000. The largest holds a seven-file kit aimed at FortiGate 100E firewalls, 26 Adaptix gopher builds and 22 Windows beacons. Some builds are named for DRBD storage and Tomcat servers. Separately, 71 modular C2 endpoints sit on 68 hosts; 8 are confirmed. Block the 7 published controllers, check firewall integrity offline, and watch Linux storage and Tomcat hosts.

INSIDE THIS CHAPTER

Three Decisions for Leaders 04

The Study in Numbers 05

01 Executive Summary

Perimeter firewalls rarely run endpoint agents. That makes them a quiet first step into a network. This report shows what one operator staged for that path: a small control kit for a FortiGate model, and implants named for the servers they would land on next. It also maps ten modular C2 and relay families seen in the same window, and separates the few confirmed controllers from the many leads.

1. **One staging server pairs a FortiGate 100E kit with 48 implant builds.**
   Workbench 1, in the United States, exposes 77 files. A `/fg100e/` folder holds a Python controller, a Node.js stage, a 212-byte second stage and two controller logs. Next to it sit 26 Adaptix gopher builds and 22 Windows beacon builds, each with its own SHA-256 hash.

2. **Linux builds are named for the servers they target next.**
   All 17 Linux amd64 gopher builds are 6,430,904 bytes, yet every hash differs. Their names point at DRBD replicated storage nodes, Apache Tomcat servers, a back-office host and a relay. Two smaller workbenches in Finland and China stage a 23.08 MB Sliver implant and an SSRF callback server.

3. **Only 8 of 71 modular C2 endpoints are confirmed controllers.**
   Six Nemo controllers on port 5000, one Xalgorix node on port 80 and one AtlasX node on port 8000 are confirmed. The other 63 endpoints are suspected or candidate listeners and stay on watchlists. Nothing links the C2 fleets to the three workbenches.

Three Decisions for Leaders

1. **Check FortiGate integrity out of band and limit appliance egress.** Verify firmware and file system integrity on internet-facing FortiGate gateways using vendor guidance. Block outbound HTTP and custom-port traffic from appliance management networks to unapproved cloud addresses.
2. **Extend visibility to Linux storage and Tomcat servers.** These tiers often run without endpoint agents. Add process and binary logging there, and alert on unsigned Go or ELF binaries that start next to replication or servlet services.
3. **Block what is confirmed, and hunt the rest.** Block the 7 published controller addresses now. Feed the shared TLS fingerprints and the port 8080 and 2334 relay patterns into correlation rules, not into block lists.

The Study in Numbers

Number	What it is
71	Modular C2 and relay endpoints across ten families, on 68 hosts
8	Confirmed controllers: 6 Nemo, 1 Xalgorix, 1 AtlasX (7 addresses published)
3	Open staging servers on port 8000 in the US, Finland and China, holding 133 files
77	Files on Workbench 1, including the seven-file `/fg100e/` kit
26	Adaptix gopher builds with unique hashes across six OS and CPU targets
22	Windows beacon builds: HTTP, DNS, service, shellcode and IAT-obfuscated
23.08 MB	Sliver implant staged on Workbench 2

How to use this report.

Security leaders can read Chapter 1 and Chapter 7. Firewall, Linux storage and middleware teams should read Chapters 3 and 4. SOC and detection engineers should start with Chapters 7 to 9. CTI analysts will find the fleet detail in Chapter 5 and the competing explanations in Chapter 6.

02

CHAPTER 02

From the Edge to the Core

IN SHORT

The files on three port 8000 servers fall into four groups: an edge kit, role-named gopher agents, Windows beacons with a Sliver implant, and modular C2 fleets. Together they describe a path from a perimeter firewall to internal storage, as a model only.

INSIDE THIS CHAPTER

The ten C2 and relay families 08

Where the endpoints run 08

02 From the Edge to the Core

Perimeter appliances sit outside normal endpoint detection. An operator who gains a foothold there can move toward Linux middleware, replicated storage and Windows domains. When such tooling sits in an open web directory, file names, sizes and build diversity show how the work is divided.

We read the staged files as a capability model. The files show what was prepared. They do not show that any step reached a target. We saw no logs from target systems.

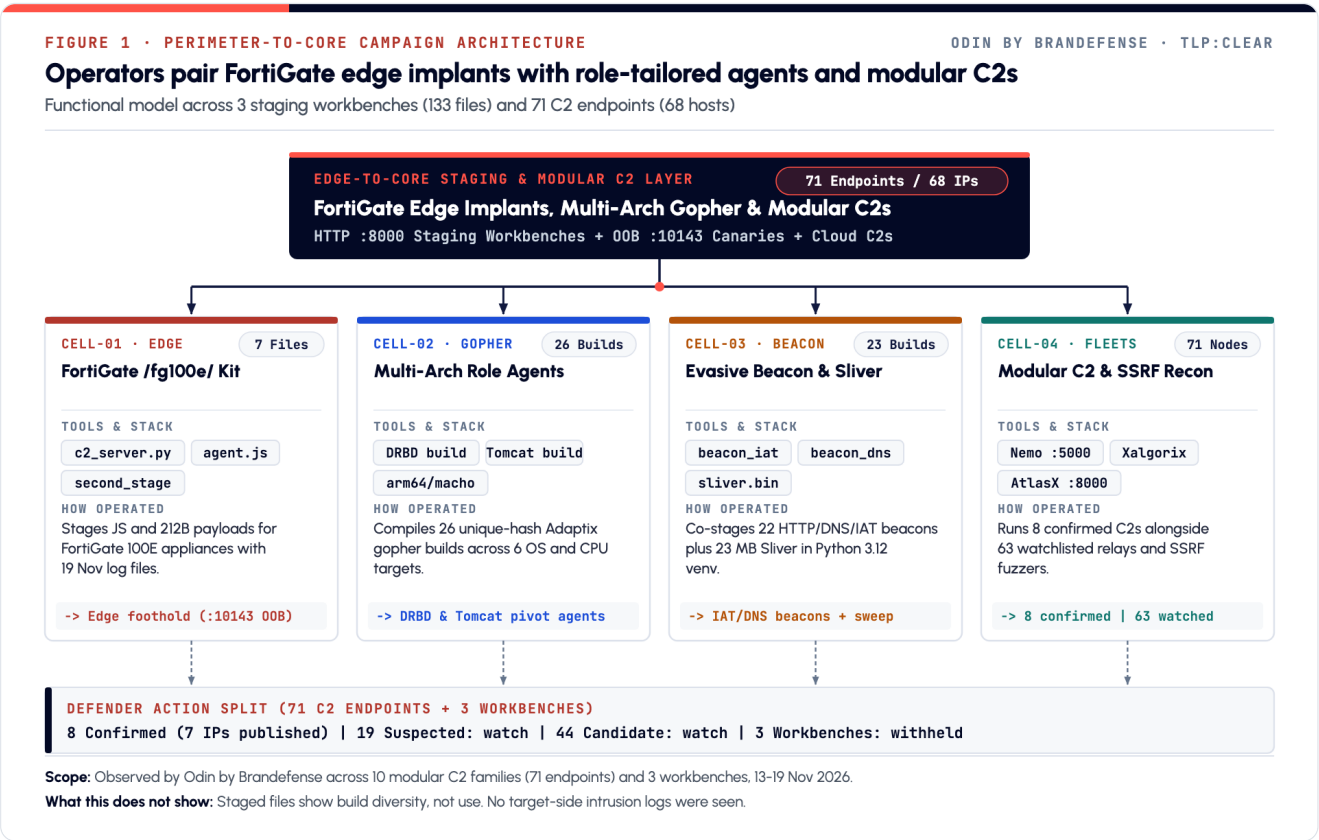


Figure 1: Operators pair FortiGate edge implants with role-tailored agents and modular C2s

The four groups in Figure 1 are an analytical model. They reflect file roles, not proven teams or separate actors.

Group	What it holds	Where
Edge	Seven-file `/fg100e/` kit for FortiGate 100E firewalls, plus out-of-band (OOB) canary listeners	Workbench 1
Gopher	26 Adaptix gopher builds across six OS and CPU targets, several named for server roles	Workbench 1
Beacon and Sliver	22 Windows beacon builds on Workbench 1, and a 23.08 MB Sliver implant on Workbench 2	Workbenches 1 and 2
Fleets	71 modular C2 and relay endpoints, plus an SSRF callback and fuzzing server	68 hosts, and Workbench 3

The ten C2 and relay families

The 71 endpoints sit on 68 distinct IPv4 hosts. Three hosts each listen on two ports within one family. No host carries two families.

Family	Role	Endpoints (hosts)	Confirmed	Suspected	Candidate
CobaltStrikeUnverified	Beacon and RAT staging	15 (14)	0	15	0
Oktos	Edge and proxy relays	12 (12)	0	0	12
GhostFrame	Edge and proxy relays	10 (10)	0	0	10
Xalgorix	Modular C2	10 (8)	1	1	8
StrikeAgent	Modular implant candidates	7 (7)	0	0	7
Nemo	Modular C2	6 (6)	6	0	0
AtlasX	Modular C2	4 (4)	1	0	3
PrismX	Beacon and RAT staging	3 (3)	0	2	1
PupyC2	Beacon and RAT staging	2 (2)	0	1	1
CloudSorcererAPT	Modular implant candidates	2 (2)	0	0	2
Total		**71 (68)**	**8**	**19**	**44**

Family names are the labels of the matching signatures. They are not attribution.

Where the endpoints run

China hosts 23 endpoints (32.4%), the United States 15 (21.1%), Hong Kong 8 (11.3%), Germany 6 (8.5%) and Ireland 5 (7.0%). The leading networks are Hangzhou Alibaba (11), Shenzhen Tencent (8), Amazon.com (8), BL Networks (6) and Hetzner Online (5).

All three workbenches serve files with a Python HTTP server on TCP 8000. The confirmed AtlasX controller also listens on port 8000. A shared default port is not a link between operators.

CHAPTER 03

Workbench 1: The FortiGate 100E Kit and the Build Matrix

IN SHORT

Workbench 1 in the United States exposes 77 files. A seven-file `/fg100e/` kit sits next to 26 gopher builds, 22 beacon builds and 22 helper scripts and logs for out-of-band callbacks, WebSocket command bridges and Windows sweeps.

INSIDE THIS CHAPTER

The `/fg100e/` kit 11

Out-of-band and lateral helpers 11

Role-named gopher builds 12

Windows beacon builds 12

A modeled path from edge to storage 12

03 Workbench 1: The FortiGate 100E Kit and the Build Matrix

Workbench 1 shows how one operator prepared payloads for a mixed enterprise network. It does not hold one generic implant. It holds separate builds for edge appliances, Linux storage and web servers, macOS endpoints, and x64 and ARM64 Windows hosts. A campaign subfolder repeats several of the builds.

Content	Files
`/fg100e/` FortiGate 100E kit	7
Adaptix gopher builds	26
Windows beacon builds	22
OOB canary, WebSocket and lateral sweep helpers and logs	22
Total	**77**

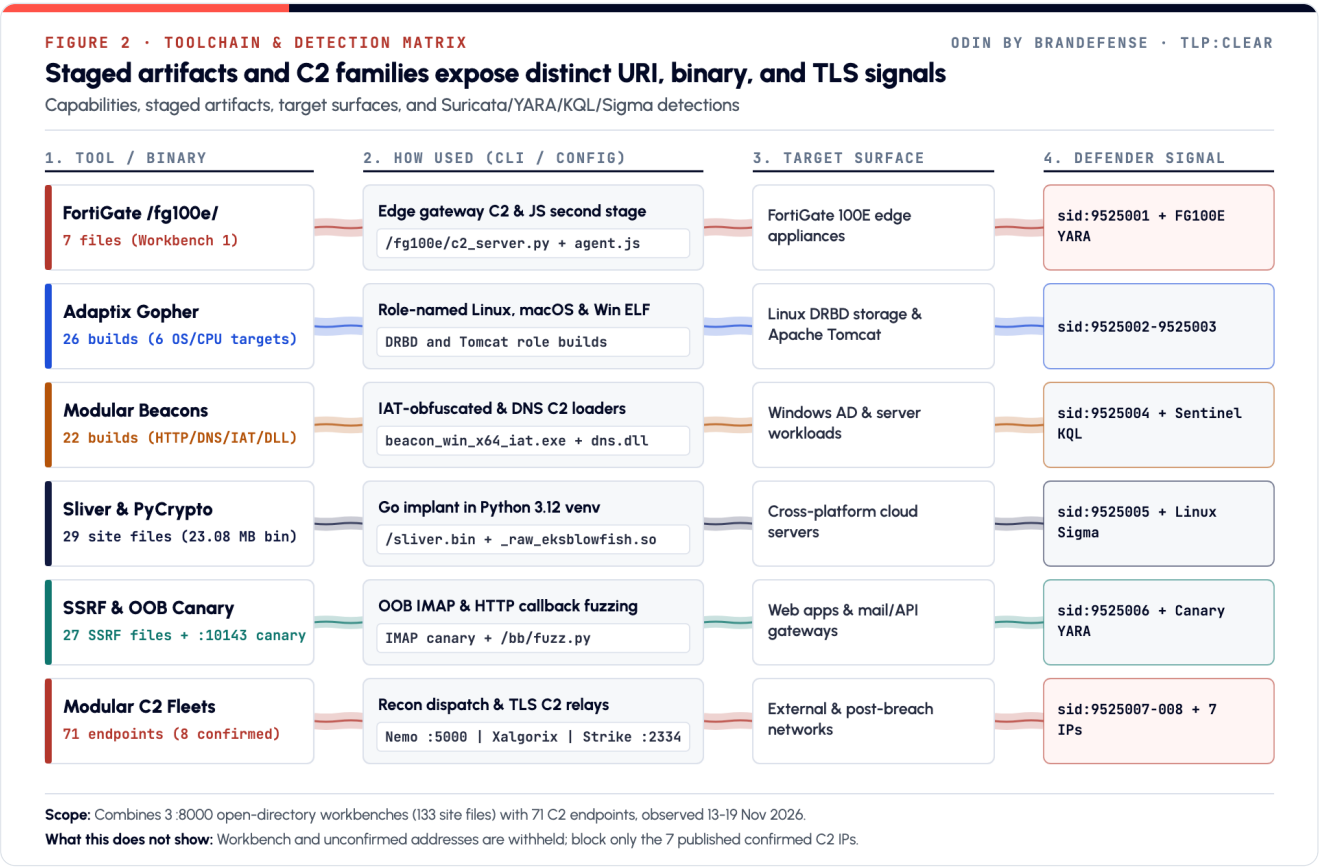


Figure 2: Staged artifacts and C2 families expose distinct URI, binary, and TLS signals

The `/fg100e/` kit

The `/fg100e/` folder holds a small controller and a staged implant chain for the FortiGate 100E (`FG-100E`) model. FortiOS combines custom Linux daemons with Node.js management components, and appliances do not run standard EDR. Small script and shellcode stages keep the disk footprint low.

Staged file	Size (bytes)	Role
`/fg100e/c2_server.py`	5,198	Python controller for implant check-ins
`/fg100e/agent.js`	744	Small JavaScript stage for a Node.js runtime
`/fg100e/start.sh`	296	Startup wrapper that writes output to `c2_stdout.log` (194 bytes)
`/fg100e/second_stage.bin`	212	Compact second-stage bootstrap
Two controller logs	101 and 194	File names carry the date 19 November 2026

The logs show that the controller wrote output. They do not show which devices, if any, checked in. Any FortiGate that contacted this controller would be a victim, and we publish no device data. Fortinet publishes FortiOS integrity and hardening guidance through its PSIRT; this data does not show which vulnerability, if any, the kit relies on.

Out-of-band and lateral helpers

At the web root sit helper scripts for callbacks and lateral movement. Out-of-band listeners let an operator confirm that a target can reach the internet, even when no reply comes back on the main channel.

Helper	Size (bytes)	Role
IMAP canary listener script (name withheld)	2,204	Listens on TCP 10143 and writes callbacks to `oob_listener_10143.out` (423 bytes)
`/execws.py`	1,929	WebSocket command bridge, paired with `oob_listener.py` (1,485 bytes)
PowerShell sweep script (name withheld)	3,010	Internal subnet and Windows host discovery, paired with a 102-byte batch file
`oob_9001.log`, `oob_8890.log`, `oob_9123.out`	Small	Logs from listeners on ports 9001, 8890 and 9123

Role-named gopher builds

The 26 gopher binaries are Go agents from the open-source AdaptixC2 framework. Each binary has a unique SHA-256 hash, even where sizes match. That pattern suggests a per-listener key or a per-target profile stamped in at build time.

Gopher build	Builds	Size (bytes)	Examples
Linux amd64 (`.elf` , `.bin`)	17	6,430,904	Role-named builds (names withheld)
Windows amd64 (`.exe`)	3	6,048,256	`gopher_windows_amd64.exe` , `gopher_win_x64.exe`
Linux arm64 (`.elf`)	2	5,963,960	`gopher_linux_arm64.elf`
Windows arm64 (`.exe`)	2	5,523,456	`gopher_windows_arm64.exe`
macOS amd64 and arm64 (`.macho`)	2	6,209,664 and 5,854,658	`gopher_macos_amd64.macho` , `gopher_macos_arm64.macho`

Several Linux amd64 builds are named for downstream roles: a second DRBD storage node, an Apache Tomcat server, a back-office host, an environment tag and a relay. We withhold the exact names because they may reflect a target's own host naming. Build names also carry four numeric task IDs, which are withheld too.

The roles matter for defenders. DRBD (Distributed Replicated Block Device) mirrors block storage across high-availability Linux cluster nodes. These nodes often back virtualization datastores or core databases, where kernel changes and third-party EDR hooks are restricted. Apache Tomcat servers often sit in a DMZ middleware tier, between reverse proxies and internal databases.

Windows beacon builds

The 22 beacon builds are C and C++ Windows payloads. They cover HTTP, DNS, service, reflective DLL, shellcode and IAT-obfuscated variants. IAT obfuscation hides the imported Windows functions from static scanners.

Beacon build	Builds	Size (bytes)	Examples
HTTP and service, x64	11	94,720 to 169,984	`beacon_http_x64.exe` , `beacon_http_x64_svc.exe` , `beacon_win_x64_v1.exe` , `agent.x64.exe`
DNS, x64 (`.exe` , `.dll`)	4	167,424 and 166,400	`beacon_dns_x64.exe` , `beacon_dns_x64.dll`
Reflective DLL and shellcode, x64	4	103,936 and 103,935	`beacon_http_x64.dll` , `beacon_http_x64_shellcode.bin`
IAT-obfuscated x64, and HTTP x86	3	94,208 and 95,232	`beacon_win_x64_iat.exe` (94,208), `beacon_http_x86.exe` (2 builds)

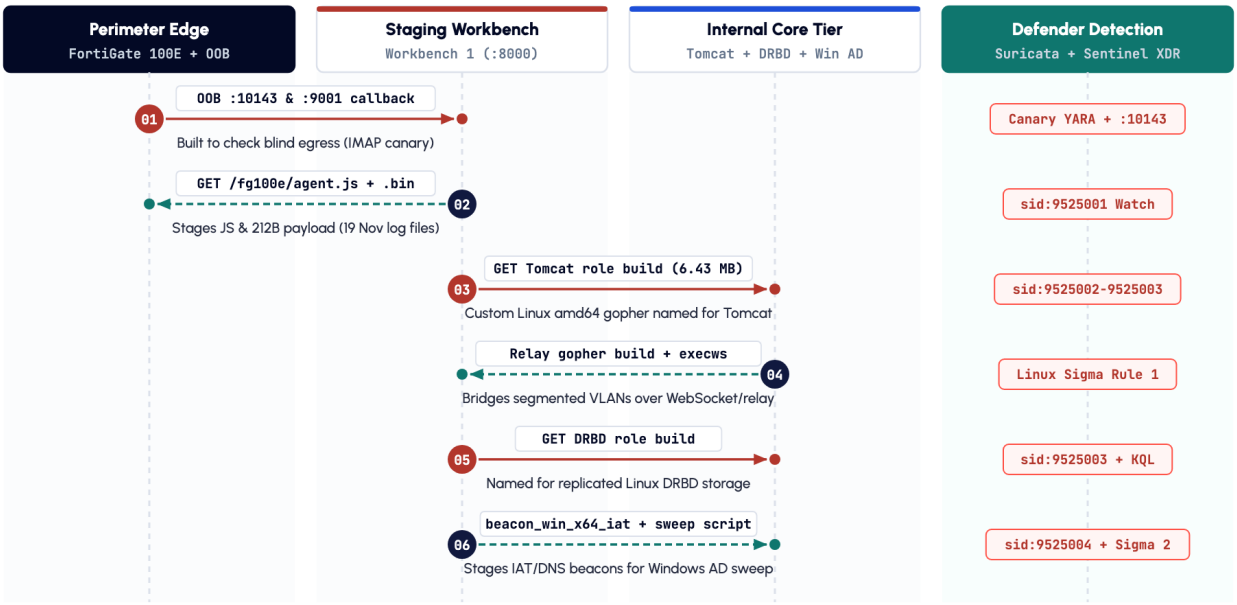
A modeled path from edge to storage

Figure 3 places the staged files on one possible path. It is an analytical model built from file names and sizes. No step was seen on a target system.

FIGURE 3 · EDGE-TO-CORE INTRUSION WORKFLOW ODIN BY BRANDEFENSE · TLP:CLEAR

Staged artifacts trace a six-step chain from FortiGate edge to DRBD storage

Analytical model of a capability sequence from 77 staged files on Workbench 1, with defender rules



Scope: Reconstructs staged tool sequence from verified filenames, sizes, and 19 Nov 2026 log names on Workbench 1 (:8000).
What this does not show: Staged design only. No step was seen on a target system.

Figure 3: Staged artifacts trace a six-step chain from FortiGate edge to DRBD storage

Step	Staged files	What they are built for	Detection
1	IMAP canary, listeners on 10143 and 9001	Check blind egress from a target	Canary YARA rule
2	`/fg100e/agent.js` and `second_stage.bin`	Stage code on a FortiGate 100E	SID 9525001
3	Tomcat-named Linux amd64 gopher (6.43 MB)	Agent for a Tomcat server	SIDs 9525002 and 9525003
4	Relay gopher build and `execws.py`	Bridge segmented networks over WebSocket or relay	Linux Sigma rule 1
5	DRBD-named Linux amd64 gopher	Agent for a DRBD storage node	SID 9525003 and KQL
6	`beacon_win_x64_iat.exe` and the sweep script	Windows beacons and domain discovery	SID 9525004 and Sigma rule 2

04

CHAPTER 04

Workbenches 2 and 3: Sliver and SSRF Callbacks

IN SHORT

Two smaller port 8000 servers, active from 16 to 18 November 2026, hold different tools. Workbench 2 in Finland stages a 23.08 MB Sliver implant inside a Python 3.12 crypto environment. Workbench 3 in China runs an SSRF callback listener and fuzzing scripts.

INSIDE THIS CHAPTER

Workbench 2: Sliver implant and a Python 3.12 crypto stack 15

Workbench 3: SSRF callback listener and rsync polling 16

04 Workbenches 2 and 3: Sliver and SSRF Callbacks

Keeping implants, crypto helpers and callback listeners on separate servers means one takedown does not stop the whole workflow. Nothing in the data shows whether one operator runs all three workbenches.

Workbench	Country, dates	Files	Main content	Detection
Workbench 1	US, 19 Nov	77	`/fg100e/` kit, 26 gopher, 22 beacon, 10143 OOB canary	SIDs 9525001 to 9525004, YARA
Workbench 2	FI, 16 to 18 Nov	29	23.08 MB `/sliver.bin`, `/HollowGhost.png`, Python 3.12 PyCryptodome	SID 9525005, Linux Sigma rule 1
Workbench 3	CN, 16 to 18 Nov	27	`/listener.py` SSRF sink, 16.43 MB log, `/bb/fuzz.py`, `/rsyncgrab/loop.sh`	SID 9525006, canary YARA rule

Workbench 2: Sliver implant and a Python 3.12 crypto stack

At the web root sits `/sliver.bin` (23,081,076 bytes). Its size is consistent with a full Go-compiled Sliver implant with built-in mutual TLS and HTTP(S) transports. Sliver is a public adversary-emulation framework that intrusion operators also use.

The root also holds `/HollowGhost.png` (146,651 bytes), whose purpose is not established, and an exposed `.git` pack file (163,894 bytes). A Python virtual environment holds native PyCryptodome libraries.

Library	Size (bytes)
`_raw_eksblowfish.abi3.so`	214,520
`_scrypt.abi3.so`	202,872
`_curve25519.abi3.so`	130,112
`_raw_aesni.abi3.so`	106,376
`_chacha20.abi3.so`	93,280

This is a standard Python crypto stack. Custom payload encoders and C2 staging proxies often use it.

Workbench 3: SSRF callback listener and rsync polling

Workbench 3 runs on a GPU-equipped Linux cloud instance. It works as a server-side request forgery (SSRF) callback and recon server. In an SSRF test, a target server is tricked into calling an address the attacker controls, and the attacker logs the call.

File	Size (bytes)	Role
`/listener.py`	971	Python 3.12 `http.server` listener started from `/tmp/listener.py`; answers HTTP 200 and logs each request
`ssrf.log`	323,215	Callback log
`ssrf_srv.log`	173,172	Server log; tracebacks show external connections reset mid-handshake (`ConnectionResetError`)
`http8000.log`	16,431,425	HTTP access log on port 8000
`/bb/fuzz.py`	1,253	Iterative parameter fuzzing
`/bb/final.py`	1,500	Verification step after fuzzing
`/rsyncgrab/loop.sh`	354	Loop that keeps polling `rsync` shares on exposed backup or storage services

The logs hold requests from third-party systems. Their contents are withheld.

05

CHAPTER 05

Modular C2 Fleets: Nemo, Xalgorix, AtlasX and Relays

IN SHORT

Ten modular C2 and relay families account for 71 endpoints on 68 hosts. Eight are confirmed controllers. The other 63 are suspected or candidate listeners, and some of them share TLS fingerprints with cloud load balancers.

INSIDE THIS CHAPTER

Confirmed controllers 18

Watched relay and implant clusters 19

Why the Cobalt Strike matches are not blocked 19

05 Modular C2 Fleets: Nemo, Xalgorix, AtlasX and Relays

These fleets keep long-running interactive C2 and recon tasks apart from the open file servers. They listen on ports 5000, 8080, 8443, 2334 and 443.

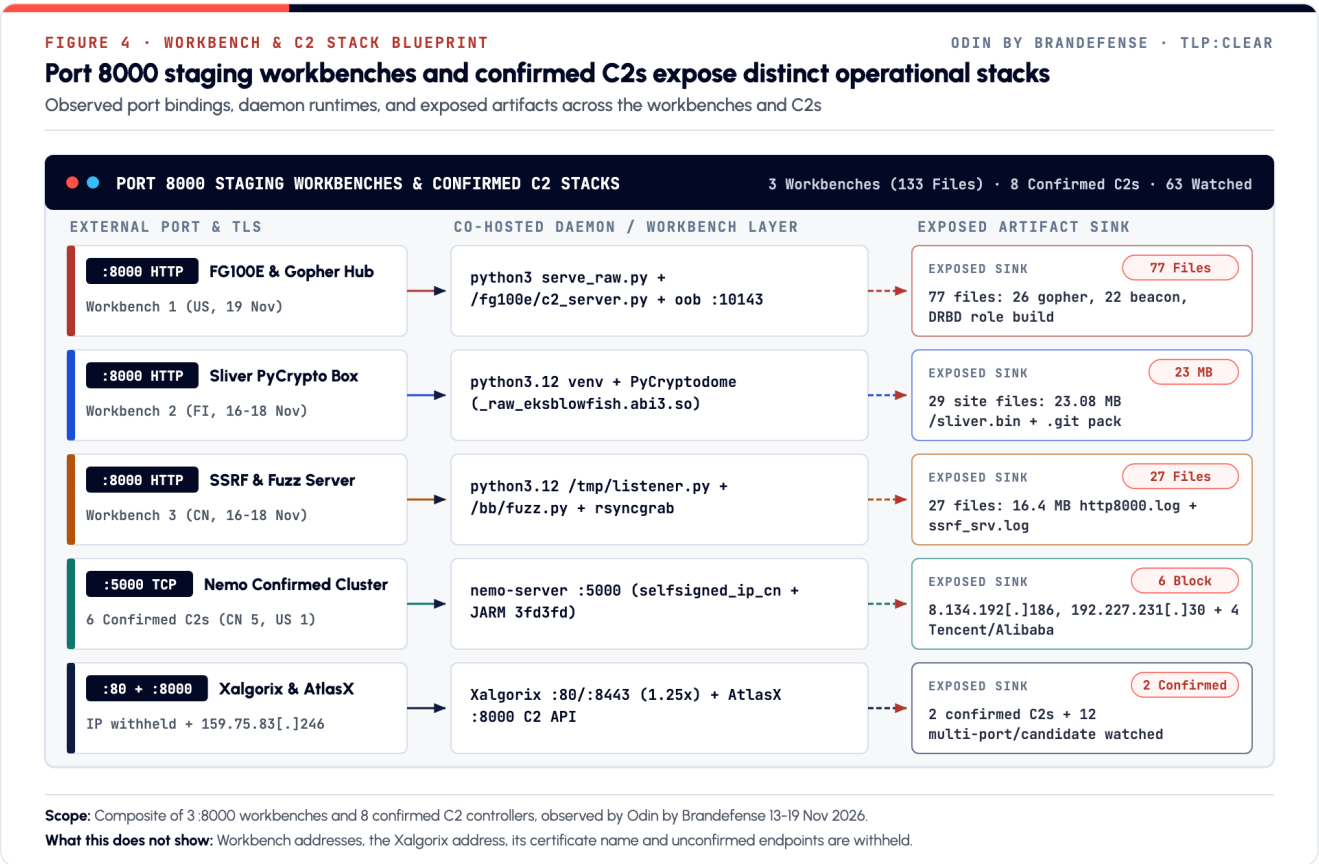


Figure 4: Port 8000 staging workbenches and confirmed C2s expose distinct operational stacks

Confirmed controllers

All eight confirmed controllers belong to three modular frameworks on rented cloud hosting.

Nemo. Six recon and C2 task controllers listen on TCP 5000. Three sit on Shenzhen Tencent, two on Hangzhou Alibaba and one on HostPapa in the United States. Five of the six share the JARM value `3fd3fd00000000000043d3fd3fd43d79451d8c63b099acafdbabb24551d0e6`. The sixth, 8.134.192[.]186, presents the same `selfsigned_ip_cn` certificate label.

Xalgorix. One confirmed node listens on port 80 on a Microsoft Corporation network in the Netherlands. Its custom certificate names a pentest-themed subdomain, so we withhold its address and certificate name (see Chapter 6). Nine companion endpoints are watched. They sit on Contabo, Hetzner Online (three on port 49567) and Data Hub Pvt. Ltd. in Nepal (one host on ports 11015 and 11080). Two separate Contabo hosts use port 8443. A candidate shares the Nemo JARM, and a suspected node shares a JARM with PupyC2 (see the watched-pattern table below).

AtlasX. One confirmed controller, 159.75.83[.]246:8000, sits on Shenzhen Tencent. Three candidates are watched: two on port 8000 (Huawei Cloud and Alibaba Cloud) and one on port 9852.

Endpoint	Family	Country and network	Fingerprint	Seen
8.134.192[.]186:5000	Nemo	CN, Hangzhou Alibaba Advertising	`selfsigned_ip_cn`	18 Nov 2026
82.156.139[.]85:5000	Nemo	CN, Shenzhen Tencent	`selfsigned_ip_cn`, JARM `3fd3fd...51d0e6`	13 Nov 2026
49.235.137[.]92:5000	Nemo	CN, Shenzhen Tencent	`selfsigned_ip_cn`, JARM `3fd3fd...51d0e6`	13 Nov 2026
192.227.231[.]30:5000	Nemo	US, HostPapa	`selfsigned_ip_cn`, JARM `3fd3fd...51d0e6`	13 Nov 2026
39.104.88[.]50:5000	Nemo	CN, Hangzhou Alibaba Advertising	`selfsigned_ip_cn`, JARM `3fd3fd...51d0e6`	13 Nov 2026
43.136.55[.]111:5000	Nemo	CN, Shenzhen Tencent	`selfsigned_ip_cn`, JARM `3fd3fd...51d0e6`	13 Nov 2026
Address withheld, port 80	Xalgorix	NL, Microsoft Corporation	Custom certificate (withheld)	13 Nov 2026
159.75.83[.]246:8000	AtlasX	CN, Shenzhen Tencent	Confirmed port 8000 C2 API	13 Nov 2026

Watched relay and implant clusters

The 63 watched endpoints are leads, not block targets. Four patterns help hunters.

Pattern	Endpoints	Detail
Port 8080 edge relays	Oktos 12, GhostFrame 10	8 of 12 Oktos nodes and all 10 GhostFrame nodes use TCP 8080. Oktos spans HK 6, CN 3, JP 1, SG 1 and LT 1, with 4 on GNET INC. GhostFrame sits in the US: 5 on BL Networks and 4 on HOSTING INDUSTRY LIMITED
StrikeAgent on port 2334	7	All 7 candidates use TCP 2334 (6 in CN, 1 in NL). Two present the certificate `CN=StrikeAgent-AtkBrain-Flash` with JARM `3fd3fd00000000000043d43d00043dc3b2afa8a5ec09b510a8559aff7899fb`
CloudSorcererAPT label	2	Two candidates on port 443, in Indonesia and Malaysia, share JARM `20d08b20d21d20d20c42d08b20b41d4e146478463a800ecaae72d070cbd53e`. The family name is a signature label, not attribution
PrismX and PupyC2	3 and 2	PrismX has two suspected listeners on ports 80 and 8888. One PupyC2 candidate on port 9000 shares JARM `2ad2ad000...7b06eb` with a suspected Xalgorix node on port 8443. That Contabo node is not the Nemo JARM candidate

Why the Cobalt Strike matches are not blocked

Fifteen endpoints match an unverified Cobalt Strike signature. None is confirmed. Seven of them sit on Amazon.com (5 in Ireland, 1 in the US, 1 in Japan) and share one JARM value, `2ad2ad16d2ad2ad00042d42d00000061256d32ed7779c14686ad100544dc8d`.

Default HTTP and HTTPS profiles and TLS cipher orderings often collide with AWS load balancers, CloudFront and Hetzner reverse proxies. Blocking AWS addresses on a JARM or header match alone would disrupt legitimate SaaS traffic. These 15 stay on a watchlist until a malleable-C2 URI or a process signal confirms them.

06

CHAPTER 06

Who Runs This: Competing Explanations

IN SHORT

The staged files fit an intrusion operator, but they also fit an authorized red team. Nothing links the three workbenches to the C2 fleets.

06 Who Runs This: Competing Explanations

We weighed three explanations for Workbench 1 and its neighbours.

Hypothesis	For	Against	Verdict
H1: An intrusion operator stages edge and internal tooling	FortiGate kit next to role-named implants; builds named for downstream server roles; public exposure	No target-side logs; no device check-ins seen	Plausible, unproven
H2: An authorized red team or penetration test	A build named for Kali Linux, a structured `MANIFEST.txt` and numbered task folders on Workbench 1; the confirmed Xalgorix node uses a pentest-themed certificate subdomain	Leaving 48 implant binaries and C2 scripts on an open server is poor practice for a tester	Plausible
H3: One operator runs all three workbenches and the C2 fleets	All three workbenches use a Python HTTP server on port 8000, the same port as the confirmed AtlasX controller	No shared IP, certificate, redirect target or campaign token links them	Not supported

Either way, an open directory that serves live implants and C2 scripts is a risk on its own. Anyone can download the builds. Defenders should detect the file names and URIs in this report, whoever owns the server.

Because H2 stays plausible, we withhold the workbench addresses, the role-named build names, the task IDs and the file hashes. For the same reason, the Xalgorix address and its certificate name are withheld.

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 7 published controller addresses: six Nemo on port 5000 and one AtlasX on port 8000. Re-check each block after 30 days, since cloud addresses get reassigned.
- [ ] Block outbound HTTP to port 8000 and connections to port 10143 from firewall management networks.
- [ ] Put suspected listeners on a 7-day watchlist and candidates on a 72-hour watchlist. Candidates on dynamic telecom networks such as CHINANET belong on a 24-hour list.

Within 72 hours

- [ ] Verify firmware and file system integrity on internet-facing FortiGate gateways, including any FG-100E units, using vendor PSIRT guidance and an offline method. Cross-check firmware against the CISA Known Exploited Vulnerabilities catalog.
- [ ] Add DRBD storage nodes and Apache Tomcat servers to process logging (`auditd` or eBPF). Alert on unsigned Go or ELF binaries named `*_agent.elf` or `gopher_*`.
- [ ] Alert when a Tomcat `java` process starts `curl` or `wget`, or connects out to ports 8000 or 10143.

Within 7 to 30 days

- [ ] Deploy Suricata SIDs 9525001 to 9525008 on perimeter and DMZ segments.
- [ ] Enable the Sentinel KQL hunt and the three Sigma rules. Fill the allowlist with the exact hostnames of your own adversary-emulation and canary hosts.
- [ ] Add the Nemo JARM `3fd3fd...51d0e6`, the StrikeAgent certificate subject and the shared JARM values to external attack surface hunting.
- [ ] Close unauthenticated `rsync` (TCP 873) exposure, and inspect web and API egress for SSRF callbacks to external port 8000 and 10143 listeners.

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. Every rule file loads without errors in its engine (Suricata 7.0.12, YARA-X, pySigma, Kusto.Language); the rules were not replayed against live traffic. Test them in your own environment before production use.

Rule file	What it catches
`odin_fortigate_fg100e_and_gopher_staging.rules`	Suricata 7: eight signatures for `/fg100e/` pulls, gopher and beacon fetches, open Sliver and SSRF listings, Nemo task dispatch and the StrikeAgent certificate
`sentinel_fortigate_edge_and_multiarch_gopher_hunt.kql`	Microsoft Sentinel and Defender XDR: contact with confirmed controllers, staging-port pulls and gopher, beacon, Sliver or `/fg100e/` process activity
`sigma_multiarch_gopher_beacon_and_edge_staging.yml`	Sigma (3 rules): Linux gopher and Sliver execution, Windows beacon execution, and `/fg100e/`, canary and SSRF script starts
`odin_fg100e_gopher_beacon_iat_artifacts.yar`	YARA (4 rules): `/fg100e/` kit manifests, multi-arch gopher build sets, beacon kits, and OOB canary or SSRF workbench files

How the Suricata signatures work

SID	Looks for	Why
9525001	Outbound `GET` whose URI starts with `/fg100e/` on ports 8000, 8080 or 80	Catches a pull of the FortiGate kit stages
9525002	Outbound `GET` for `gopher_linux`, `gopher_windows`, `gopher_macos`, `gopher_win`, `gopher_kali` or `gopher_relay`	Catches multi-arch gopher fetches
9525003	Outbound `GET` for a URI ending in `<name>_agent.elf`	Catches role-named Linux agents without naming a target
9525004	Outbound `GET` for `beacon_dns`, `beacon_http` or `beacon_win` with `x64` or `x86`	Catches beacon payload fetches
9525005	Port 8000 response with `Directory listing for /` and `sliver.bin`	Catches an open Sliver staging directory
9525006	Port 8000 response with `Directory listing for /` and `ssrf_srv.log`	Catches an open SSRF callback server
9525007	Outbound JSON `POST` to `/v1/task/` on port 5000	Catches Nemo task dispatch
9525008	TLS certificate subject `CN=StrikeAgent-AtkBrain-Flash` from port 2334	Hunt only: the listeners are candidates. Pair with endpoint checks before acting

How the other rules work

The KQL query has a 7-day lookback. It flags devices that contact the 7 confirmed controller addresses. It also flags `curl`, `wget`, Python, Node.js, PowerShell, `rundll32` or `java` processes that reach the staging and C2 ports in this report. A process branch flags files named `*_agent.elf`, `gopher_*`, `beacon_*` or `sliver.bin`, and command lines that reference `/fg100e/`, `second_stage.bin` or the canary and WebSocket scripts. A third condition flags a device that reaches 10 or more remote addresses on two or more of these ports. The allowlist never hides a match on a confirmed controller.

The first Sigma rule flags Linux execution of `*_agent.elf`, default gopher builds or `/sliver.bin`. The second flags Windows beacon and gopher executables, and command lines that load beacon DLL or shellcode files. The third flags Python, Node.js or shell processes that start the `/fg100e/` scripts, the IMAP canary, `execws.py`, `oob_listener.py`, `/bb/fuzz.py` or `rsyncgrab/loop.sh`. Each rule excludes only exact hostnames that you register.

YARA rule	Fires when
`ODIN_FortiGate_FG100E_PostExploit_C2_Kit`	`/fg100e/` or `c2_server.py` appears with `second_stage.bin`, `agent.js` or `c2_stdout.log`
`ODIN_Adaptix_MultiArch_Gopher_Build_Set`	A manifest or listing names gopher builds for three or more OS and CPU targets
`ODIN_Beacon_DNS_HTTP_IAT_Staging_Manifest`	Two beacon build names appear with `execws.py` or `oob_listener.py`
`ODIN_OOB_IMAP_Canary_and_SSRF_Fuzz_Workbench`	The IMAP canary and its port 10143 output file appear together, or two SSRF workbench file names do

09

CHAPTER 09

Indicators

IN SHORT

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

INSIDE THIS CHAPTER

MITRE ATT&CK 29

09 Indicators

Appendix A holds 7 published indicators: 7 addresses and 0 hashes. They are the confirmed Nemo and AtlasX controllers listed in Chapter 5.

Type	Count	Where
Confirmed controller endpoints to block (IP and port)	7	Appendix A and Chapter 5
Confirmed controller with address withheld (Xalgorix)	1	Text only
Hashes of selected staged files (SHA-256)	7	Withheld, like every other build hash
Watched C2 and relay endpoints	63	Withheld; described by family, port and fingerprint

Supporting fingerprints stay in the text: the Nemo JARM `3fd3fd00000000000043d3fd3fd43d79451d8c63b099acafdbabb24551d0e6`, the StrikeAgent certificate subject `CN=StrikeAgent-AtkBrain-Flash`, the URI patterns in Chapter 8, and the file names and sizes in Chapters 3 and 4.

Withheld: the three workbench addresses, the seven selected file hashes and all other build hashes, the role-named build names and task IDs, the Xalgorix address and certificate name, and all suspected or candidate addresses.

MITRE ATT&CK

The staging servers serve tools for transfer into a target network, which maps to T1105 Ingress Tool Transfer. SIDs 9525001 to 9525006 and the YARA rules detect this transfer and its staging directories.

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, with data to 19 November. Workbench 1 files were captured on 19 November. Workbenches 2 and 3 were seen from 16 to 18 November. Log dates are file names, not proof of activity.
- **Staged is not used.** The workbenches show what was prepared. We saw no target-side logs, and no step of the modeled chain was observed on a victim.
- **Owner not established.** An authorized red team is a plausible owner of Workbench 1. Treat its builds as hostile tooling for detection, without assuming hostile intent.
- **Confirmed is not attribution.** Family names come from signature labels. They do not tie these hosts to any named group or published campaign.
- **Leads stay leads.** The 63 watched endpoints rest on weaker signals. Do not block them without a second signal.
- **No links between populations.** Nothing ties the workbenches to the C2 fleets, or the fleets to each other.
- **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. CISA. Known Exploited Vulnerabilities Catalog. https://www.cisa.gov/known-exploited-vulnerabilities-catalog (verified 20 November 2026)
2. PSIRT advisories for FortiOS and FortiGate. https://www.fortiguard.com/psirt (verified 20 November 2026)
3. MITRE ATT&CK. T1105 Ingress Tool Transfer. https://attack.mitre.org/techniques/T1105/ (verified 20 November 2026)
4. LINBIT. DRBD 9 User Guide. https://linbit.com/drbd-user-guide/drbd-guide-9_0-en/ (verified 19 November 2026)
5. Apache Software Foundation. Apache Tomcat 10.1 Security Considerations. https://tomcat.apache.org/tomcat-10.1-doc/security-howto.html (verified 20 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 20 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 20 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 (7) 35

B. Detection rules (4 files) 36

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
R25-IOC-01	IPv4	43.136.55[.]111	5000	nemo	BLOCK
R25-IOC-02	IPv4	39.104.88[.]50	5000	nemo	BLOCK
R25-IOC-03	IPv4	192.227.231[.]30	5000	nemo	BLOCK
R25-IOC-04	IPv4	49.235.137[.]92	5000	nemo	BLOCK
R25-IOC-05	IPv4	82.156.139[.]85	5000	nemo	BLOCK
R25-IOC-06	IPv4	8.134.192[.]186	5000	nemo	BLOCK
R25-IOC-07	IPv4	159.75.83[.]246	8000	atlasx	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_fg100e_gopher_beacon_iat_artifacts.yar

YARA

```
/*
   Odin by Brandefense - Monograph 25 / 30 YARA Artifact Suite
   Scope: Configuration, manifest, listing and staging controller artifacts for
          FortiGate /fg100e/ staging kits, multi-arch Adaptix gopher build sets,
          modular HTTP/DNS/IAT beacon kits, and OOB IMAP / SSRF fuzzing workbenches.
   Classification: TLP:CLEAR
   Compiles in YARA-X. Test in your environment before production use.
*/

rule ODIN_FortiGate_FG100E_PostExploit_C2_Kit
{
    meta:
        author = "Odin by Brandefense"
        monograph = "25 / 30"
        classification = "TLP:CLEAR"
        category = "staging_kit"
        description = "Matches FortiGate 100E (/fg100e/) post-exploitation controller and stage manifests or
listings"
    strings:
        $fg1 = "/fg100e/" ascii
        $fg2 = "c2_server.py" ascii
        $st1 = "second_stage.bin" ascii
        $st2 = "agent.js" ascii
        $lg1 = "c2_stdout.log" ascii
    condition:
        filesize < 256KB and ($fg1 or $fg2) and ($st1 or $st2 or $lg1)
}

rule ODIN_Adaptix_MultiArch_Gopher_Build_Set
{
    meta:
        author = "Odin by Brandefense"
        monograph = "25 / 30"
        classification = "TLP:CLEAR"
        category = "multiarch_staging"
        description = "Matches manifests or listings that index gopher builds for three or more OS/CPU targets"
    strings:
        $a1 = "gopher_linux_amd64" ascii
        $a2 = "gopher_linux_arm64.elf" ascii
        $a3 = "gopher_macos_amd64.macho" ascii
        $a4 = "gopher_macos_arm64.macho" ascii
        $a5 = "gopher_windows_amd64.exe" ascii
        $a6 = "gopher_windows_arm64.exe" ascii
    condition:
        filesize < 512KB and 3 of ($a*)
}

rule ODIN_Beacon_DNS_HTTP_IAT_Staging_Manifest
{
    meta:
        author = "Odin by Brandefense"
        monograph = "25 / 30"
        classification = "TLP:CLEAR"
        category = "beacon_kit"
        description = "Matches modular Windows HTTP, DNS, shellcode, and IAT-evasive beacon staging manifests
next to helper scripts"
    strings:
        $b1 = "beacon_win_x64_iat.exe" ascii
        $b2 = "beacon_dns_x64.dll" ascii
        $b3 = "beacon_http_x64_shellcode.bin" ascii
```

```
        $h1 = "execws.py" ascii
        $h2 = "oob_listener.py" ascii
    condition:
        filesize < 512KB and 2 of ($b*) and 1 of ($h*)
}

rule ODIN_OOB_IMAP_Canary_and_SSRF_Fuzz_Workbench
{
    meta:
        author = "Odin by Brandefense"
        monograph = "25 / 30"
        classification = "TLP:CLEAR"
        category = "recon_callback"
        description = "Matches OOB IMAP (:10143) canary listeners and SSRF callback fuzzing workbench artifacts"
    strings:
        $o1 = "imap_canary_" ascii
        $o2 = "oob_listener_10143" ascii
        $s1 = "ssrf_srv.log" ascii
        $s2 = "/tmp/listener.py" ascii
        $s3 = "/bb/fuzz.py" ascii
        $s4 = "rsyncgrab/loop.sh" ascii
    condition:
        filesize < 1024KB and (($o1 and $o2) or (2 of ($s*)))
}
```

odin_fortigate_fg100e_and_gopher_staging.rules

SURICATA

```
# Odin by Brandefense - Monograph 25 / 30 Suricata 7.x Detection Suite
# Coverage: FortiGate /fg100e/ staging kit, multi-arch gopher, modular beacons, Sliver/SSRF workbenches, Nemo and
StrikeAgent
# Classification: TLP:CLEAR
# Loads in Suricata 7.0.12. Test in your environment before production use.

alert http $HOME_NET any -> $EXTERNAL_NET [8000,8080,80] (msg:"ODIN [R25] FortiGate 100E (/fg100e/) Post-
Exploitation Stage Pull"; flow:established,to_server; http.method; content:"GET"; http.uri; content:"/fg100e/";
startswith; fast_pattern; classtype:trojan-activity; sid:9525001; rev:2; metadata:created_at 2026_11_19,
updated_at 2026_11_20, deployment perimeter, tactic command_and_control;)

alert http $HOME_NET any -> $EXTERNAL_NET [8000,8080,80] (msg:"ODIN [R25] Multi-Architecture Adaptix Gopher
Implant Staging Fetch"; flow:established,to_server; http.method; content:"GET"; http.uri; content:"gopher_";
fast_pattern; pcre:"/\/gopher_(linux|windows|macos|win|kali|relay)[^\/]*$/i"; classtype:trojan-activity; sid:
9525002; rev:2; metadata:created_at 2026_11_19, updated_at 2026_11_20, deployment perimeter, tactic
command_and_control;)

alert http $HOME_NET any -> $EXTERNAL_NET [8000,8080,80] (msg:"ODIN [R25] Role-Named Linux Agent ELF Retrieval";
flow:established,to_server; http.method; content:"GET"; http.uri; content:"_agent.elf"; endswith; fast_pattern;
pcre:"/\/[A-Za-z0-9]+_agent\.elf$/"; classtype:trojan-activity; sid:9525003; rev:2; metadata:created_at
2026_11_19, updated_at 2026_11_20, deployment perimeter, tactic lateral_movement;)

alert http $HOME_NET any -> $EXTERNAL_NET [8000,8080,80] (msg:"ODIN [R25] Modular HTTP DNS or IAT Beacon Payload
Staging Request"; flow:established,to_server; http.method; content:"GET"; http.uri; content:"beacon_";
fast_pattern; pcre:"/\/beacon_(dns|http|win)_(x64|x86)/"; classtype:trojan-activity; sid:9525004; rev:2;
metadata:created_at 2026_11_19, updated_at 2026_11_20, deployment perimeter, tactic defense_evasion;)

alert http $EXTERNAL_NET 8000 -> $HOME_NET any (msg:"ODIN [R25] Open-Directory Sliver Implant Staging Listing";
flow:established,to_client; http.stat_code; content:"200"; http.response_body; content:"Directory listing for /";
content:"sliver.bin"; fast_pattern; classtype:trojan-activity; sid:9525005; rev:2; metadata:created_at
2026_11_19, updated_at 2026_11_20, deployment perimeter, tactic resource_development;)

alert http $EXTERNAL_NET 8000 -> $HOME_NET any (msg:"ODIN [R25] Open-Directory SSRF Callback and Fuzzing
Workbench Listing"; flow:established,to_client; http.stat_code; content:"200"; http.response_body;
content:"Directory listing for /"; content:"ssrf_srv.log"; fast_pattern; classtype:attempted-recon; sid:9525006;
rev:2; metadata:created_at 2026_11_19, updated_at 2026_11_20, deployment perimeter, tactic reconnaissance;)

alert http $HOME_NET any -> $EXTERNAL_NET 5000 (msg:"ODIN [R25] Nemo Recon and C2 Controller API Dispatch";
flow:established,to_server; http.method; content:"POST"; http.uri; content:"/v1/task/"; fast_pattern;
http.content_type; content:"application/json"; classtype:trojan-activity; sid:9525007; rev:2; metadata:created_at
2026_11_19, updated_at 2026_11_20, deployment perimeter, tactic reconnaissance;)
```

```
alert tls $EXTERNAL_NET 2334 -> $HOME_NET any (msg:"ODIN [R25] StrikeAgent-AtkBrain-Flash TLS Certificate (hunt,
candidate listener)"; flow:established,to_client; tls.cert_subject; content:"CN=StrikeAgent-AtkBrain-Flash";
fast_pattern; classtype:trojan-activity; sid:9525008; rev:2; metadata:created_at 2026_11_19, updated_at
2026_11_20, deployment perimeter, tactic command_and_control;)
```

sentinel_fortigate_edge_and_multiarch_gopher_hunt.kql

KQL (SENTINEL / DEFENDER XDR)

```
// Odin by Brandefense - Monograph 25 / 30 Microsoft Sentinel & Defender XDR Hunt
// Correlates contact with confirmed modular C2 controllers, staging-port pulls, and gopher/beacon/Sliver/fg100e
process activity
// Classification: TLP:CLEAR
// Parses with Kusto.Language. Test in your environment before production use.

let Lookback = 7d;
// Exact hostnames of your own registered adversary-emulation and canary hosts. Never use wildcards here.
let ApprovedInfraHosts = dynamic(["SEC-VALIDATION-LNX-01", "REDTEAM-WIN-LAB-01", "DECEPTION-CANARY-01"]);
let ConfirmedC2IPs = dynamic([
    "8.134.192.186", "82.156.139.85", "49.235.137.92", "192.227.231.30",
    "39.104.88.50", "43.136.55.111", "159.75.83.246"
]);
let ModularStagingAndC2Ports = dynamic([2334, 5000, 8000, 8080, 8443, 8444, 9000, 9852, 10143, 11443, 49567]);
let NetHits = DeviceNetworkEvents
    | where Timestamp >= ago(Lookback)
    | where RemoteIP in (ConfirmedC2IPs)
        or (RemotePort in (ModularStagingAndC2Ports) and InitiatingProcessFileName in~ ("curl", "wget", "python",
"python3", "node", "powershell.exe", "rundll32.exe", "java"))
    | summarize
        NetFirstSeen = min(Timestamp),
        NetLastSeen = max(Timestamp),
        DistinctRemoteIPs = dcount(RemoteIP),
        DistinctPorts = make_set(RemotePort, 16),
        MatchedC2IPs = make_set_if(RemoteIP, RemoteIP in (ConfirmedC2IPs), 16),
        NetProcesses = make_set(InitiatingProcessFileName, 16),
        ConnectionCount = count()
        by DeviceId, DeviceName;
let ProcHits = DeviceProcessEvents
    | where Timestamp >= ago(Lookback)
    | where FileName endswith "_agent.elf"
        or FileName startswith "gopher_"
        or FileName startswith "beacon_"
        or FileName =~ "sliver.bin"
        or ProcessCommandLine matches regex @"/fg100e/|second_stage\.bin|execws\.py|oob_listener\.py|
imap_canary_|beacon_http_x64_shellcode"
    | summarize
        ProcFirstSeen = min(Timestamp),
        StagedFiles = make_set(FileName, 16),
        StagedCommandLines = make_set(ProcessCommandLine, 8)
        by DeviceId, DeviceName;
NetHits
| join kind=fullouter (ProcHits) on DeviceId
| extend DeviceId = coalesce(DeviceId, DeviceId1), DeviceName = coalesce(DeviceName, DeviceName1)
| extend C2Hit = array_length(MatchedC2IPs) > 0, ProcHit = array_length(StagedFiles) > 0
| extend BurstHit = DistinctRemoteIPs >= 10 and array_length(DistinctPorts) >= 2
| where C2Hit == true or ProcHit == true or BurstHit == true
// The allowlist never hides a match on a confirmed C2 address.
| where C2Hit == true or not(DeviceName in~ (ApprovedInfraHosts))
| extend AlertDisposition = case(
    C2Hit == true, "BLOCK_CONFIRMED_MODULAR_C2",
    ProcHit == true, "ISOLATE_STAGED_EDGE_OR_GOPHER_IMPLANT",
    "HUNT_MODULAR_STAGING_OR_OOB_CANARY"
)
| project NetFirstSeen, NetLastSeen, ProcFirstSeen, DeviceName, AlertDisposition, MatchedC2IPs,
DistinctRemoteIPs, DistinctPorts, NetProcesses, StagedFiles, StagedCommandLines, ConnectionCount
| sort by ConnectionCount desc
```

sigma_multiarch_gopher_beacon_and_edge_staging.yml

SIGMA

```
title: Role-Named Multi-Architecture Gopher or Sliver Implant Execution on Linux Servers
id: 8c2f41a9-3d70-4b18-9e42-202611192501
status: experimental
description: Detects execution of role-named Adaptix gopher ELF agents (for example builds named for DRBD
storage, Tomcat or relay roles), default multi-arch gopher builds, or staged Sliver implants on Linux hosts.
Converts with pySigma; test before production use.
author: Odin by Brandefense
date: 2026/11/19
modified: 2026/11/20
tags:
  - attack.execution
  - attack.command_and_control
  - attack.lateral_movement
logsource:
  category: process_creation
  product: linux
detection:
  selection_implant_bin:
    Image|endswith:
      - '_agent.elf'
      - '/gopher_linux_amd64.elf'
      - '/gopher_linux_arm64.elf'
      - '/sliver.bin'
  selection_implant_cli:
    CommandLine|re: '([A-Za-z0-9]+_agent\.elf|gopher_(relay|linux_amd64|linux_arm64|kali)|/sliver\.bin)'
  filter_lab:
    # Placeholder host names: replace with your own approved hosts.
    Computer:
      - 'SEC-VALIDATION-LNX-01'
  condition: (selection_implant_bin or selection_implant_cli) and not filter_lab
falsepositives:
  - Authorized red-team infrastructure registered in a security validation inventory
  - Unrelated software that ships a binary ending in _agent.elf
level: high
---
title: Modular HTTP DNS or IAT-Obfuscated Beacon Execution on Windows
id: 4b7e90c3-6a12-4f85-8d29-202611192502
status: experimental
description: Identifies execution of IAT-obfuscated, DNS, HTTP or service Windows beacons and multi-architecture
gopher Windows builds, or loading of beacon DLL and shellcode files. Converts with pySigma; test before
production use.
author: Odin by Brandefense
date: 2026/11/19
modified: 2026/11/20
tags:
  - attack.defense_evasion
  - attack.command_and_control
  - attack.discovery
logsource:
  category: process_creation
  product: windows
detection:
  selection_exec:
    Image|endswith:
      - '\beacon_win_x64_iat.exe'
      - '\beacon_dns_x64.exe'
      - '\beacon_http_x64.exe'
      - '\beacon_http_x64_svc.exe'
      - '\beacon_win_x64_svc.exe'
      - '\gopher_windows_amd64.exe'
      - '\gopher_windows_arm64.exe'
      - '\gopher_win_x64.exe'
  selection_loader:
    CommandLine|contains:
      - 'beacon_dns_x64.dll'
      - 'beacon_http_x64.dll'
      - 'beacon_http_x64_shellcode.bin'
  filter_redteam:
```

```
    # Placeholder host names: replace with your own approved hosts.
    Computer:
      - 'REDTEAM-WIN-LAB-01'
  condition: (selection_exec or selection_loader) and not filter_redteam
falsepositives:
  - Sanctioned adversary-emulation lab workstations
level: high
---
title: FortiGate Edge Staging Controller and Out-of-Band Canary Listener Spawn
id: 2e9a65d4-1c83-47f0-a631-202611192503
status: experimental
description: Detects staging controllers and stages from the FortiGate 100E (/fg100e/) kit, out-of-band IMAP
canary listeners, WebSocket execution bridges, or SSRF fuzzing scripts started by a script runtime on Linux.
Converts with pySigma; test before production use.
author: Odin by Brandefense
date: 2026/11/19
modified: 2026/11/20
tags:
  - attack.resource_development
  - attack.command_and_control
  - attack.reconnaissance
logsource:
  category: process_creation
  product: linux
detection:
  selection_runtime:
    Image|endswith:
      - '/python'
      - '/python3'
      - '/node'
      - '/bash'
      - '/sh'
  selection_script:
    CommandLine|contains:
      - '/fg100e/c2_server.py'
      - '/fg100e/agent.js'
      - 'imap_canary_'
      - 'oob_listener.py'
      - 'execws.py'
      - '/bb/fuzz.py'
      - 'rsyncgrab/loop.sh'
  filter_canary:
    # Placeholder host names: replace with your own approved hosts.
    Computer:
      - 'DECEPTION-CANARY-01'
  condition: selection_runtime and selection_script and not filter_canary
falsepositives:
  - Internal deception canary listeners registered with security operations
level: high
```

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/fortigate-edge-kits-and-role-named-implants/
Short Read · brandefense.io/odin-research/blog/a-fortigate-kit-staged-next-to-implants-named-for-storage/
Defense Kit · brandefense.io/odin-research/defense-kit/