ODIN

TLP:CLEAR

ODIN RESEARCH · MONOGRAPH 29 / 30 · COMMODITY MALWARE

Mobile Overlay C2s, Cloud Worms and Recon-Phishing Fleets

How HookBot, P2PInfect, EvilGoPhish and dual-use recon suites split into block and watch decisions

Two-week observation window ending 18 November 2026 · Data cutoff 19 November 2026 · Prepared November 2026

Full Report · Public edition · Version 1.1

717
Endpoints across ten families, on 687 distinct hosts

1
HookBot controller corroborated in the window and published for blocking

16
Older HookBot entries (11) and Octopus panels (5) kept on watchlists, addresses withheld

667
ARL, ScopeSentry and reNgine recon endpoints, all kept on watchlists (93.0%)

What Is in This Report. 12 chapters and two appendices.

How HookBot, P2PInfect, EvilGoPhish and dual-use recon suites split into block and watch decisions

01	Executive Summary	04
02	Four Layers of Infrastructure	07
03	Android Banking Panels	10
04	P2PInfect and Moobot Cloud Worms	13
05	EvilGoPhish and the Recon Suites	15
06	An Exposed Agent Workbench	18
07	Block or Watch: Telling Hostile from Dual-Use	21
08	What Defenders Should Do	23
09	Detection Rules	25
10	Indicators	28
11	Scope and Limits	30
12	Sources	32
A	Indicators of compromise (1)	34
B	Detection rules (4 files)	35

IN SHORT

In a two-week observation window ending 18 November 2026, Odin by Brandefense mapped 717 endpoints on 687 hosts across ten infrastructure families. Android banking panels, cloud worms, reverse-proxy phishing kits and recon suites sit side by side. Recon suites make up 93.0% of the endpoints. Only one HookBot controller, 152.70.24[.]49, is corroborated in the window as adversary infrastructure. Block it, keep everything else on watchlists, and close public Redis access.

01

CHAPTER 01

Executive Summary

IN SHORT

In a two-week observation window ending 18 November 2026, Odin by Brandefense mapped 717 endpoints on 687 hosts across ten infrastructure families. Android banking panels, cloud worms, reverse-proxy phishing kits and recon suites sit side by side. Recon suites make up 93.0% of the endpoints. Only one HookBot controller, 152.70.24[.]49, is corroborated in the window as adversary infrastructure. Block it, keep everything else on watchlists, and close public Redis access.

INSIDE THIS CHAPTER

Three Decisions for Leaders 04

The Study in Numbers 05

01 Executive Summary

This report covers four kinds of internet-facing infrastructure. Android banking trojan panels steal credentials and control phones. Cloud worms spread through exposed Redis and SSH services. Phishing kits proxy login pages to steal session cookies. Recon suites map an organization's external attack surface. The same tools serve attackers and defenders, so the key question for each group is whether to block or to watch.

1. **One HookBot controller is the only address published for blocking.**
   152.70.24[.]49 on Oracle Cloud in Germany is a HookBot command-and-control (C2) server seen in the window. Eleven more HookBot addresses are older feed entries, kept on watchlists. Five Octopus panels on adjacent IBM Cloud addresses fit a lab and go to review.

2. **P2PInfect peers and EvilGoPhish servers stay on watchlists, not block lists.**
   Sixteen P2PInfect nodes, 15 on Alibaba Cloud, use high gossip ports from 60103 to 60140. These are most likely infected third-party servers. Nine EvilGoPhish servers pair a GoPhish admin port with a reverse proxy, but authorized phishing tests use the same kit.

3. **Dual-use recon suites dominate the volume and stay on watch.**
   ARL (488 endpoints), ScopeSentry (138) and reNgine (41) make up 667 endpoints. Security teams run these suites too, so none is blocked. Separately, one exposed workbench in Japan held 1,725 files, including AI agent logs and an authentication archive.

Three Decisions for Leaders

1. **Block the one confirmed controller and watch everything else.** Put 152.70.24[.]49 on perimeter and mobile VPN block lists. Route the older HookBot entries, the Octopus cluster, the 667 recon endpoints and all worm and phishing nodes to time-bound watchlists.
2. **Require phishing-resistant sign-in and protect mobile apps.** FIDO2 passkeys defeat reverse-proxy session theft. Android restricted settings and runtime integrity checks limit the Accessibility abuse that HookBot, Daam and L3MON depend on.
3. **Close public Redis and audit open file servers.** Turn on Redis protected mode and authentication, and keep it off the internet. Remove public directory listings from engineering and automation servers.

The Study in Numbers

Number	What it is
717	Endpoints across ten families, on 687 distinct hosts
1	HookBot controller corroborated in the window and published for blocking
16	Older HookBot entries (11) and Octopus panels (5) kept on watchlists, addresses withheld
667	ARL, ScopeSentry and reNgine recon endpoints, all kept on watchlists (93.0%)
16	P2PInfect peer nodes; 15 sit on Alibaba Cloud
9	EvilGoPhish phishing servers; 4 use the default admin port 3333
1,725	Files on one exposed AI agent workbench in Japan
24	Hosts that bind several services, including 2 that run both ARL and ScopeSentry

How to use this report.

Security leaders can read Chapter 1 and Chapter 7. Detection engineers should start with Chapters 7 to 10. CTI analysts will find the infrastructure detail in Chapters 2 to 6.

02

CHAPTER 02

Four Layers of Infrastructure

IN SHORT

The 717 endpoints fall into four functional layers: recon, phishing, worms and mobile C2. Recon suites are large and noisy; mobile panels and worm meshes are small and compact.

INSIDE THIS CHAPTER

Counts by family 08

Where the endpoints run 08

02 Four Layers of Infrastructure

Defenders who ingest raw internet scan feeds face an imbalance. Recon frameworks produce a large volume of banners on ports such as 5003 and 8082. Mobile banking panels and worm meshes run in small, fast-changing clusters. Treating both alike either floods analysts with false positives or blocks benign security tools.

We group the ten families into four layers. This grouping is an analytical model. Nothing observed links the operators of one layer to another.

Layer	Families	Endpoints
Recon	ARL, ScopeSentry, reNgine	667
Initial access phishing	EvilGoPhish	9
Cloud and edge worms	P2PInfect, Moobot	18
Mobile C2	HookBot, Octopus, Daam, L3MON	23

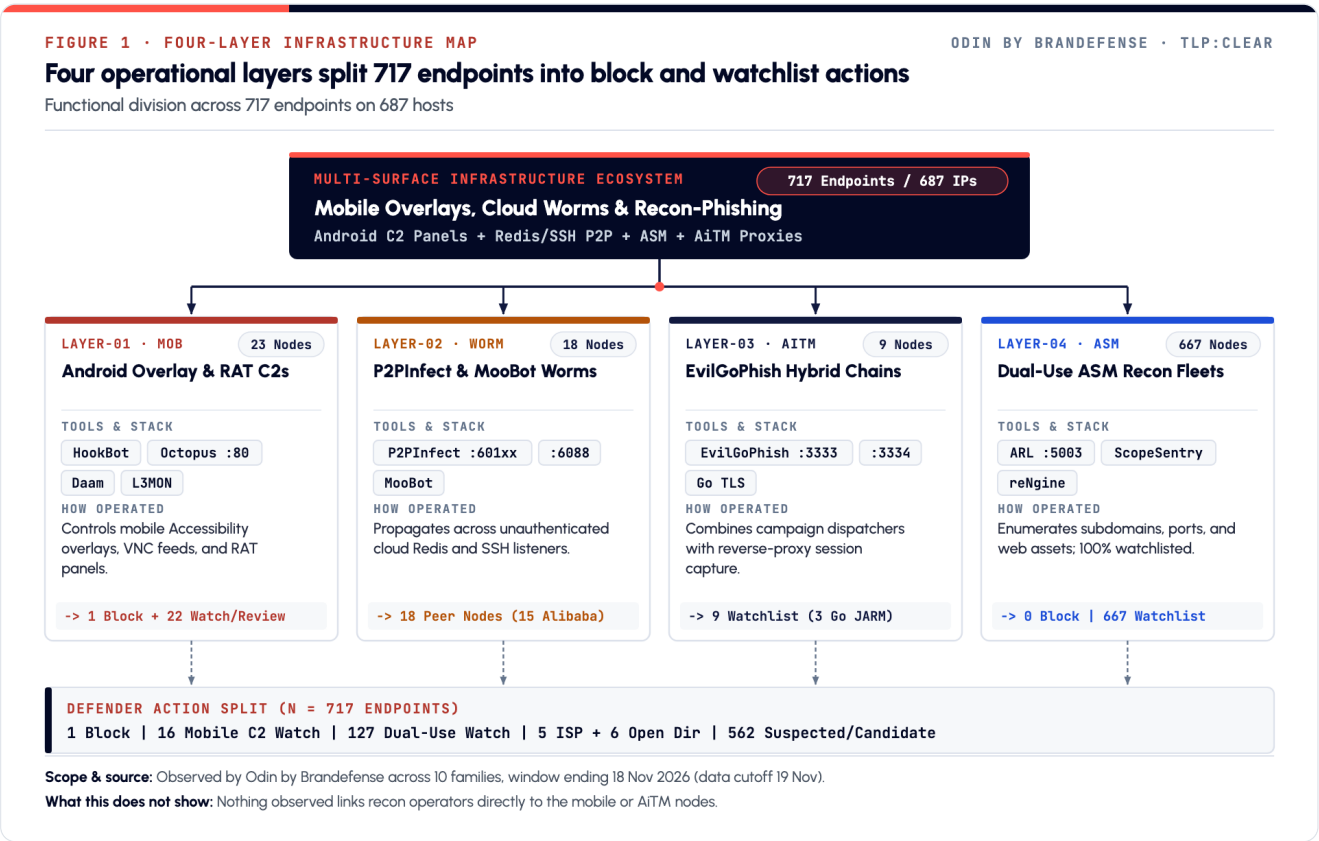


Figure 1: Four operational layers split 717 endpoints into block and watchlist actions

Counts by family

Each endpoint is an IP and port pair. The 717 endpoints sit on 687 hosts, because two servers run two families each.

Family	Role	Endpoints	Hosts	Block	Watch
ARL	Dual-use recon	488	474	0	488
ScopeSentry	Dual-use recon	138	136	0	138
reNgine	Dual-use recon	41	29	0	41
P2PInfect	Cloud peer-to-peer worm	16	16	0	16
HookBot	Android overlay C2	13	13	1	12
EvilGoPhish	Reverse-proxy phishing	9	9	0	9
Octopus	C2 panel	5	5	0	5
Daam	Android overlay and RAT	3	3	0	3
L3MON	Android NodeJS RAT	2	2	0	2
Moobot	Edge botnet worm	2	2	0	2

Where the endpoints run

Six countries lead.

Country	Endpoints	Share
China	332	46.3%
United States	164	22.9%
Hong Kong	81	11.3%
Germany	30	4.2%
Singapore	25	3.5%
Japan	14	2.0%

The layers use different hosting. ARL runs mostly in China (284), the United States (128) and Hong Kong (52). ScopeSentry shows a similar spread: China 43, Hong Kong 28 and the United States 20. The largest recon hosting networks are Shenzhen Tencent (155 rows), SonderCloud (91) and Hangzhou Alibaba (82).

The HookBot and Octopus panels sit on European and global VPS providers instead. IBM Cloud hosts five in Germany, Contabo two in Germany, Hostinger two, and Oracle one in Germany. P2PInfect keeps 12 of its 16 nodes (75.0%) on Alibaba (US) Technology in Singapore.

03

CHAPTER 03

Android Banking Panels

IN SHORT

HookBot, Octopus, Daam and L3MON account for 23 endpoints. Only one HookBot controller is corroborated in the window; the other 22 endpoints stay on watchlists.

INSIDE THIS CHAPTER

HookBot 10
Octopus 11
Daam and L3MON 11

03 Android Banking Panels

Modern Android banking trojans combine three tricks. A WebView overlay draws a fake login screen over a banking app. Android `AccessibilityService` lets the malware read the screen and tap buttons. Real-time VNC screen streaming lets the operator move money directly on the phone.

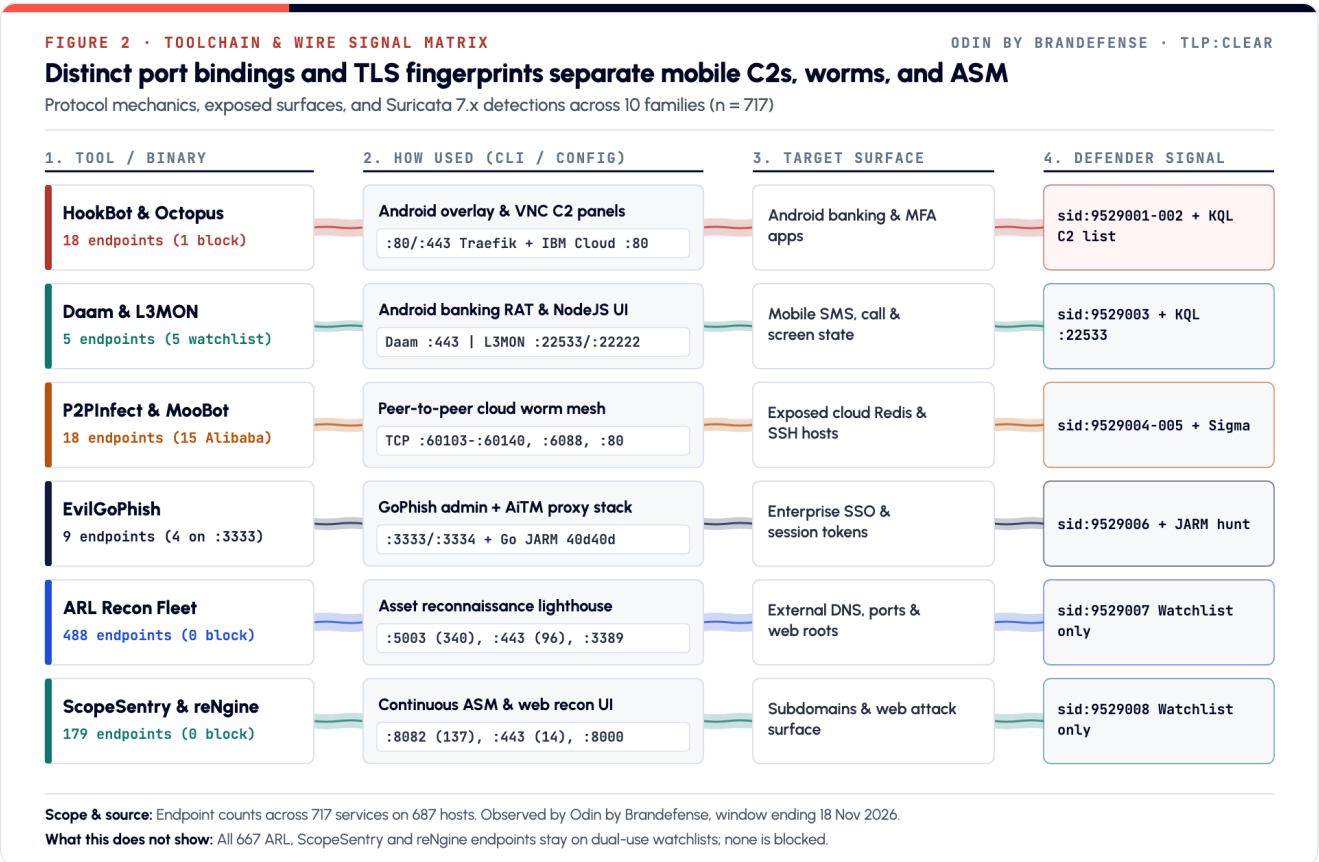


Figure 2: Distinct port bindings and TLS fingerprints separate mobile C2s, worms and recon suites

HookBot

HookBot has 13 endpoints. One controller, 152.70.24[.]49 on port 80, is corroborated in the window. It sits on Oracle Corporation in Germany and presents a certificate for a custom domain, which is withheld.

Eleven more HookBot addresses come from older feed entries. None of them has a record inside this window. Cloud addresses get reassigned, so these entries go to watchlists and their addresses are withheld. The thirteenth endpoint listens on port 2223 and stays a watchlist candidate.

The older entries still carry useful fingerprints. Eight listen on port 80 and four on port 443 across the 12 controller records. Five present the stock `TRAEFIK DEFAULT CERT` certificate, which points to a containerized deployment behind the Traefik reverse proxy.

Fingerprint	Records	Hosting
`TRAEFIK DEFAULT CERT`, JARM prefix `3fd3fd...`	3	Contabo GmbH (DE, 2), RouterHosting LLC (US)
`TRAEFIK DEFAULT CERT`, JARM prefix `26d26d...`	1	The Constant Company (CL)
`TRAEFIK DEFAULT CERT`, JARM prefix `29d29d...`	1	Amazon.com (ID)
JARM prefix `0003fd...`	1	Hostinger (LT)

Hunt the certificate subject together with the JARM values. Treat a match as a lead, not as proof of a live controller.

Octopus

Five Octopus panels listen on port 80 inside adjacent IBM Cloud subnets in Frankfurt, Germany. Four form a contiguous block and the fifth sits in a nearby subnet. All five were seen on 13 November 2026.

Octopus is also the name of an open-source red-team C2 framework. Five adjacent cloud addresses fit a test lab as well as an attacker. We therefore withhold the addresses and send the cluster to analyst review rather than to block lists.

Daam and L3MON

Daam and L3MON are not corroborated yet, so they stay on watchlists. Their addresses are withheld.

Family	Endpoints	Ports	Hosting
Daam	3	443	Saudi Arabia, United States and India; one US node presents `CN=Plesk` on IONOS
L3MON	2	22533, 22222	Amazon (US) and Iliad Italia (IT)

L3MON is an open-source NodeJS remote-management panel for Android devices. Its default port 22533 is a useful hunting signal by itself. Both L3MON candidates were seen on 19 November 2026.

CHAPTER 04

P2PInfect and Moobot Cloud Worms

IN SHORT

P2PInfect builds a peer-to-peer mesh out of exposed Redis and SSH servers. Its 16 nodes are most likely infected third-party machines, so this report treats them as victims and publishes no addresses.

INSIDE THIS CHAPTER

What the P2PInfect nodes look like 13
Why no addresses are published 13
Moobot 13

04 P2PInfect and Moobot Cloud Worms

P2PInfect does not rely on one central server. It scans for exposed Redis instances and SSH servers, then turns infected hosts into peers that talk over high ports. Unit 42 describes it as a Rust-based, self-replicating worm. Taking down one node does not stop the mesh.

What the P2PInfect nodes look like

Feature	Count
Nodes in total	16
On Alibaba Cloud	15 (93.8%): 12 in Singapore, 2 in China, 1 in Hong Kong
Gossip ports 60103 to 60140	12
Port 6088	2
Port 8281	1
Port 9684	1
Self-signed `CN=gateway` (shared JARM prefix `3fd3fd...`)	2
Self-signed `CN=localhost`	3

Several nodes were seen on 19 November 2026, on ports 60103, 60136, 60139, 60140 and 6088.

Why no addresses are published

An infected cloud server belongs to someone who is likely unaware of the infection. Blocking it would hurt its owner as much as the worm. Publishing it would expose the owner. We therefore describe the nodes by port, certificate and hosting type only.

Moobot

Two Moobot nodes serve HTTP on port 80. Moobot is a Mirai variant that stages payloads over plain HTTP. One node sits on DA International Group in Bulgaria. Both stay on watchlists with addresses withheld.

05

CHAPTER 05

EvilGoPhish and the Recon Suites

IN SHORT

Nine EvilGoPhish servers combine a GoPhish campaign panel with a reverse proxy that captures session cookies. They sit next to 667 recon endpoints that map login portals, subdomains and certificates.

INSIDE THIS CHAPTER

EvilGoPhish 15

ARL, ScopeSentry and reNgine 16

05 EvilGoPhish and the Recon Suites

Recon suites and phishing kits serve adjacent stages of an attack. Recon finds the login portals. The phishing kit then copies one of them. Figure 3 shows this sequence as a conceptual model, not as one observed intrusion.

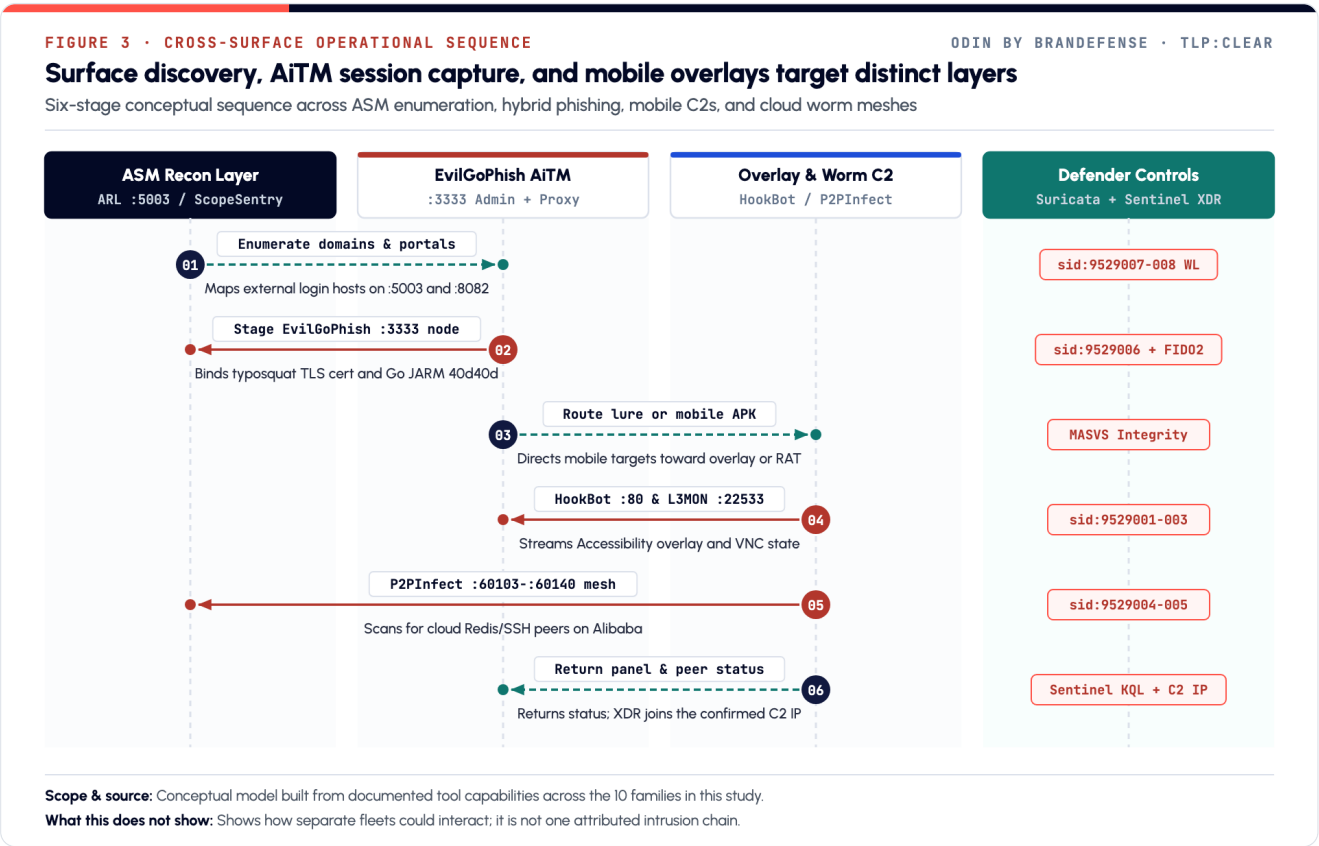


Figure 3: Surface discovery, AiTM session capture and mobile overlays target distinct layers

EvilGoPhish

EvilGoPhish joins two tools. GoPhish runs the email campaign through an admin API on port 3333. An Evilginx-style reverse proxy on port 443 relays the real login page. It captures the session cookie and the one-time MFA code as the victim signs in.

Feature	Detail
Servers	9, one per host
Countries	Germany 3, United States 2, Ireland 1, China 1, India 1, Indonesia 1
Ports	3333 (4 servers), plus 3322, 3334, 33335, 10002 and 443
Shared TLS fingerprint	3 servers share the Go `crypto/tls` JARM `40d40d40d00000000043d43d00043d6aff5ab0f4fc31897186312c9e639319`
Lookalike certificates	2 DigitalOcean servers present certificates for lookalike domains (withheld)
Activity	8 were seen on 19 November 2026; two 3333 servers, on IONOS (DE) and Amazon (IE), were first seen that day

The Go JARM value is common to many Go programs, so it is a hunting lead, not proof by itself. Pair it with port 3333 or a lookalike certificate.

Companies also run EvilGoPhish for authorized phishing exercises. A single port or JARM match cannot tell the two apart. All nine servers stay on watchlists until a lure domain confirms hostile use. Their addresses and the targeted brands are withheld.

ARL, ScopeSentry and reNgine

These three open-source attack-surface management (ASM) frameworks enumerate subdomains, ports and web assets. They make up 93.0% of the endpoints (667 of 717).

Suite	Endpoints	Hosts	Main ports	Fingerprints
ARL	488	474	5003 (340, 69.7%), 443 (96, 19.7%)	123 nodes share JARM `15d3fd16d29d29d00042d43d000000fbc10435df141b3459e26f69e76d5947`; 19 suspected hosts also expose RDP on 3389 with self-signed certificates named after internal 10.x addresses
ScopeSentry	138	136	8082 (119, 86.2%), 18082 (3), 8443 (3)	Hosted in China 43, Hong Kong 28, United States 20 and Japan 11
reNgine	41	29	8082 (18), 443 (14), 80 (4), 8000 (2)	17 present the default `CN=rengine.example[.]com`; 15 of those share JARM `2ad2ad0002ad2ad0002ad2ad2ad2ada5fc7ec76b7ed0b839211317d3844486`

reNgine has the most ports per host in this study, at 1.41 endpoints per host. One DigitalOcean stack in Germany runs it on ports 80, 443 and 8000 together.

Two Chinese servers run ARL and ScopeSentry side by side. One, on China Telecom, pairs ARL on 5003 with ScopeSentry on 8082. The other, on CHINANET BACKBONE, pairs ARL on 20000 and 43047 with ScopeSentry on 8082. Their addresses are withheld as dual-use.

CHAPTER 06

An Exposed Agent Workbench

IN SHORT

One server in Japan exposed an open directory of 1,725 files. It mixed AI agent logs, an agent backup, a Gemini authentication archive and cloud deployment notes.

INSIDE THIS CHAPTER

What the workbench held 19

Open directories on recon nodes 19

06 An Exposed Agent Workbench

Operators sometimes leave a Python file server or a reverse proxy open while they stage tools. The directory tree then shows tool pairings that a banner alone cannot. We do not know who owns this server. It may belong to a developer rather than an attacker, so its address and file paths are withheld.

FIGURE 4 · CONVERGED WORKBENCH & CO-HOSTED STACK BLUEPRINT
ODIN BY BRANDEFENSE · TLP:CLEAR

Exposed agent workbenches and co-hosted ASM stacks reveal multi-tool consolidation

Observed port bindings, runtime artifacts and open directories on one workbench and on recon hubs



Scope & source: Composite of one exposed workbench (listing dated 19 Nov 2026) and co-hosted recon servers.
What this does not show: Workbench and recon host addresses are withheld; the workbench owner is not established.

Figure 4: Exposed agent workbenches and co-hosted recon stacks show multi-tool consolidation

What the workbench held

The directory listing is dated 19 November 2026. Its main files are listed here.

Artifact	Size	What it is
Hermes agent backup archive	821.1 MB	Complete Hermes agent workspace and state
Bun runtime binary	89.0 MB	Self-contained JavaScript and TypeScript agent runtime
Repository archive (name withheld)	14.7 MB	Packaged code repository in the web root
Gemini authentication archive	12.8 MB	Compressed Gemini API or CLI authentication bundle
Linux utility binary (name withheld)	6.59 MB	Precompiled `amd64` tool
Flask service log	3.59 MB	Log of a Python Flask orchestration service
OpenClaw agent log, dated 19 November	3.43 MB	Session log of the OpenClaw autonomous agent
OpenClaw agent log, dated 20 November	292.5 KB	The next day's log, from the server's local UTC+9 clock
Headless browser log	82.1 KB	A Chrome DevTools Protocol daemon, `Headless Browser v0.1.9`
Cloud deployment notes	100.4 KB	Notes on deploying Cloudflare Workers, D1, KV and R2 with `wrangler` to a third-party domain (withheld)

The workbench also held a Hermes repository index, a backed-up proxy configuration and Cloudflare login logs. Together these show one person running autonomous agents, browser automation and cloud deployment from one machine. Model authentication files sat in an internet-readable folder.

Open directories on recon nodes

Open directories also show up inside the ARL fleet. Six ARL candidate endpoints on five hosts also serve an open HTTP file index. Three of the six also expose configuration or credential files next to the console.

Hosting (country)	Ports	Notes
Shenzhen Tencent (CN)	5003, 5030	Two ports on one host
Shenzhen Tencent (CN)	5003	Exposed configuration files
LUCIDACLOUD (HK)	5003	Exposed configuration files
LUCIDACLOUD (HK)	5003	Exposed configuration files
Tencent Building (KR)	892	Open directory

These hosts stay on watchlists for analyst review. Their addresses are withheld.

07

CHAPTER 07

Block or Watch: Telling Hostile from Dual-Use

IN SHORT

Most endpoints here run tools that defenders also use. Blocking them all would break legitimate scanners and still miss rotating worm peers, so only one corroborated controller goes on block lists.

07 Block or Watch: Telling Hostile from Dual-Use

Three innocent explanations apply to large parts of this data:

- **Sanctioned security tools.** Enterprise EASM and penetration-testing teams run ARL, ScopeSentry and reNgine.
- **Authorized phishing exercises.** Security-awareness teams deploy EvilGoPhish on cloud servers.
- **Victim machines and shared addresses.** P2PInfect peers are infected third-party servers. Some recon nodes sit on telecom subscriber networks where addresses rotate.

The table below shows how the 717 endpoints split into actions.

Group	Endpoints	Suggested action
HookBot controller corroborated in the window	1	Block; review the lease after 30 days
Older HookBot feed entries	11	Watch; re-check before any block
Octopus panels that fit a test lab	5	Analyst review
Confirmed recon suites: 94 ARL, 23 reNgine, 10 ScopeSentry	127	Watch for 7 days; never block automatically
Recon nodes on telecom subscriber networks (4 confirmed, 1 suspected)	5	Watch for 24 hours
ARL candidates with an open directory	6	Watch for 7 days; analyst review first
Suspected recon nodes: 23 ARL, 2 reNgine	25	Watch for 7 days
Uncorroborated candidates, including the 16 P2PInfect peers and 9 EvilGoPhish servers	537	Watch for 72 hours; act only on corroboration

The telecom subscriber networks are China Unicom, China Telecom, ChinaTelecom Jiangsu, China Mobile and Telekomunikasi Indonesia. Addresses there change hands, so a long block would hit later, unrelated users.

The action bar at the foot of Figure 1 shows the same split.

CHAPTER 08

What Defenders Should Do

IN SHORT

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

08 What Defenders Should Do

Within 24 hours

- [ ] Block the confirmed HookBot controller 152.70.24[.]49 on perimeter firewalls and mobile VPN gateways. Review the cloud lease after 30 days, because cloud addresses get reassigned.
- [ ] Put the recon, worm and phishing groups on time-bound watchlists, as in Chapter 7. Do not block them automatically.

Within 72 hours

- [ ] Take cloud Redis (6379) off the public internet. Turn on protected mode and ACL authentication.
- [ ] Require SSH key-only access on cloud servers, so P2PInfect cannot recruit them.
- [ ] Require FIDO2 or WebAuthn sign-in on workforce SSO portals. These methods defeat reverse-proxy session theft.
- [ ] Watch new TLS certificates for lookalikes of your own domains.

Within 7 to 30 days

- [ ] Enforce Android restricted settings and OWASP MASVS runtime integrity checks in mobile banking apps. This limits sideloaded Accessibility overlays and VNC screen capture.
- [ ] Deploy the Suricata rules and the Sentinel hunt. Keep the ARL, ScopeSentry and reNgine rules in alert mode, never in drop mode.
- [ ] Audit engineering and red-team servers for open HTTP directories that expose agent logs, backups or cloud authentication files.
- [ ] Hunt the certificate subjects and JARM values in this report across your external attack surface tooling.

09

CHAPTER 09

Detection Rules

IN SHORT

Four rule files cover the mobile panels, cloud worms, EvilGoPhish and recon suites. They hold eight Suricata signatures, one Sentinel query, three Sigma rules and four YARA rules.

INSIDE THIS CHAPTER

How the Suricata signatures work 25

How the other rules work 25

09 Detection Rules

The full rules are in Appendix B.

Rule file	What it catches
`odin_hookbot_p2pinfect_evilgophish_recon.rules`	Suricata 7: eight signatures for mobile C2 check-ins, P2PInfect and Moobot fetches, EvilGoPhish campaign calls and recon task APIs
`sentinel_mobile_overlay_p2pinfect_and_evilgophish_hunt.kql`	Microsoft Sentinel and Defender XDR: contact with the confirmed HookBot controller, and fleet ports used by Redis, Node, Python or GoPhish processes
`sigma_p2pinfect_redis_ssh_worm_and_evilgophish.yml`	Sigma (3 rules): Redis or SSH child processes with worm markers, EvilGoPhish launches, and unapproved recon workers on Linux
`odin_hookbot_l3mon_p2pinfect_evilgophish_artifacts.yar`	YARA (4 rules): HookBot or Daam panel configs, L3MON manifests, P2PInfect peer configs and EvilGoPhish configs

How the Suricata signatures work

All eight signatures inspect outbound HTTP from inside your network.

SID	Looks for	Ports
9529001	`POST` to `/api/hookbot/` with `"accessibility"` and `"vnc_state"` in the body	80, 443, 2223
9529002	`POST` to `/api/octopus/` with `"agent_uuid"` in the body	80
9529003	`POST` to `/client_login` with `"device_model"` in the body (Daam or L3MON)	443, 22222, 22533
9529004	`/p2p/peers` request with an `X-Peer-Node:` header (P2PInfect)	6088, 8281, 9684, 60103 to 60140
9529005	`/moobot.` fetch with a `Wget/` user agent	80
9529006	`POST` to `/api/campaigns/` with an `X-EvilGoPhish-Key:` header	443, 3322, 3333, 3334, 10002, 33335
9529007	`POST` to `/api/task/` with `"task_tag"` (ARL), as a policy event	443, 5003, 5004, 5080, 8443
9529008	`POST` to `/api/scan/create` with `"scan_engine"` (ScopeSentry or reNgine), as a policy event	443, 8000, 8082, 8443, 18082

SIDs 9529007 and 9529008 use the `policy-violation` class. Keep them in alert mode, and add `pass` rules for your own named EASM scanners. Self-signed 443 and 601xx channels need TLS inspection or endpoint data to be seen.

The Suricata file loads in Suricata 7.0.12. The signatures have not been replayed against captured traffic from each family, so test them in alert mode first.

How the other rules work

The KQL query looks back 7 days and has two parts. The first flags any device that contacts the confirmed HookBot controller address. This branch always alerts, even from an approved host. The second flags fleet ports such as 3333, 6088, 22533 and 60103 when `redis-server`, `python3`, `node`, `curl`, `wget` or `gophish` opens them.

It then joins process events that mention `p2pinfect`, `evilgophish`, `l3mon` or recon workers, or `slaveof` on Redis. An approved-host list holds exact hostnames for your own scanners and phishing simulators. That list never hides a confirmed controller match.

The first Sigma rule needs a parent of `redis-server` or `sshd` and a command line with a worm port, `p2pinfect` or `slaveof`. The second flags command lines that name `evilgophish`, `evilginx`, `gophish_admin` or :3333. The third flags ARL, ScopeSentry or reNgine worker launches. Each rule has an exact-hostname exclusion for sanctioned servers.

Each YARA rule is bounded to files under 512 KB and needs at least two independent strings.

YARA rule	Fires when
`ODIN_HookBot_Daam_Overlay_Panel_Config`	A HookBot or Daam marker appears with `accessibility_inject` or `vnc_state`
`ODIN_L3MON_NodeJS_Android_Panel_Manifest`	An L3MON marker, port 22533 or 22222, and `/client_login` appear together
`ODIN_P2PInfect_Cloud_Worm_Peer_Config`	`X-Peer-Node` and `/p2p/peers` appear with a gossip port or `redis_peer_sync`
`ODIN_EvilGoPhish_Hybrid_AiTM_Config`	An EvilGoPhish marker, an admin marker and `evilginx_sub` appear together

10

CHAPTER 10

Indicators

IN SHORT

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

10 Indicators

Appendix A holds 1 published indicator: 1 address and no hashes. It is the HookBot controller 152.70.24[.]49 on port 80, on Oracle Corporation in Germany.

Type	Count	Where
Confirmed controller address	1	Appendix A
Older HookBot feed entries	11	Described in Chapter 3; addresses withheld
Octopus panels that fit a test lab	5	Described in Chapter 3; addresses withheld
Other watchlist endpoints (HookBot candidate, Daam, L3MON, P2PInfect, Moobot, EvilGoPhish, recon suites)	700	Described in Chapters 3 to 5; addresses withheld
File hashes from the exposed workbench	5	Withheld

Addresses of older HookBot entries, the Octopus cluster, dual-use recon hosts, infected worm peers, phishing servers of unknown purpose and the workbench are withheld. So are the workbench file hashes, lookalike domains and targeted brands.

Supporting fingerprints: `TRAEFIK DEFAULT CERT`, `CN=gateway`, `CN=rengine.example[.]com`, and the JARM values named in Chapters 3 to 5.

11

CHAPTER 11

Scope and Limits

IN SHORT

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

11 Scope and Limits

- **Service data, not victim logs.** This report describes internet-facing services. It does not show what any panel, worm or phishing kit did to a specific victim.
- **Older HookBot entries.** For 11 of the 12 HookBot controller records, the latest record predates this window, so these addresses are withheld and kept on watchlists.
- **Octopus may be a lab.** Five adjacent IBM Cloud addresses running a panel named after an open-source red-team C2 can be a test lab. Our data cannot rule that out.
- **Layers are a model.** The four-layer grouping is analytical. No shared certificate, address or campaign token links the recon, phishing, worm and mobile groups.
- **Dual-use tools.** ARL, ScopeSentry, reNgine and EvilGoPhish serve both defenders and attackers. Our data cannot tell which use applies to a given host.
- **Worm peers are victims.** P2PInfect nodes are most likely infected servers. Their owners are not adversaries.
- **Workbench owner unknown.** The exposed workbench may belong to a developer. Its contents show tooling, not hostile action.
- **Rule testing.** 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.

12

CHAPTER 12

Sources

IN SHORT

This chapter lists the public references used in the report.

12 Sources

1. Android Developers, AccessibilityService: developer.android.com/guide/topics/ui/accessibility/service (verified on 20 November 2026).
2. OWASP Foundation, Mobile Application Security Verification Standard (MASVS): mas.owasp.org/MASVS (verified on 20 November 2026).
3. Palo Alto Networks Unit 42, P2PInfect: The Rusty Peer-to-Peer Self-Replicating Worm: unit42.paloaltonetworks.com/peer-to-peer-worm-p2pinfect (verified on 20 November 2026).
4. Redis Documentation, Redis security: redis.io/docs/latest/operate/oss_and_stack/management/security (verified on 20 November 2026).
5. CISA, Implementing Phishing-Resistant MFA (fact sheet): cisa.gov/sites/default/files/publications/fact-sheet-implementing-phishing-resistant-mfa-508c.pdf (verified on 20 November 2026).
6. OISF, Suricata 7.0 User Guide: HTTP keywords: docs.suricata.io/en/suricata-7.0.0/rules/http-keywords.html (verified on 20 November 2026).
7. Salesforce Engineering, Easily Identify Malicious Servers on the Internet with JARM: engineering.salesforce.com/easily-identify-malicious-servers-on-the-internet-with-jarm-e095edac525a (verified on 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 (1) 34

B. Detection rules (4 files) 35

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
R29-IOC-01	IPv4	152.70.24[.]49	80	hookbot	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_hookbot_l3mon_p2pinfect_evilgophish_artifacts.yar

YARA

```
/*
   Odin by Brandefense - Monograph 29 / 30 YARA Artifact Suite
   Scope: Sanitized configuration and panel manifests for HookBot/Daam,
          L3MON (:22533), P2PInfect (:601xx), and EvilGoPhish (:3333).
   Classification: TLP:CLEAR
*/

rule ODIN_HookBot_Daam_Overlay_Panel_Config
{
    meta:
        author = "Odin by Brandefense"
        monograph = "29 / 30"
        classification = "TLP:CLEAR"
        description = "Matches HookBot or Daam Android Accessibility overlay and VNC panel configuration
artifacts"
    strings:
        $hb1 = "HOOKBOT_PANEL" ascii
        $hb2 = "/api/hookbot/" ascii
        $dm1 = "daam_overlay" ascii nocase
        $ov1 = "accessibility_inject" ascii nocase
        $ov2 = "vnc_state" ascii nocase
    condition:
        filesize < 512KB and ($hb1 or $hb2 or $dm1) and ($ov1 or $ov2)
}

rule ODIN_L3MON_NodeJS_Android_Panel_Manifest
{
    meta:
        author = "Odin by Brandefense"
        monograph = "29 / 30"
        classification = "TLP:CLEAR"
        description = "Matches L3MON NodeJS Android remote management server configuration on ports 22533 or
22222"
    strings:
        $l1 = "L3MON" ascii
        $l2 = "l3mon_manager" ascii nocase
        $p1 = "22533" ascii
        $p2 = "22222" ascii
        $u1 = "/client_login" ascii
    condition:
        filesize < 512KB and ($l1 or $l2) and ($p1 or $p2) and $u1
}

rule ODIN_P2PInfect_Cloud_Worm_Peer_Config
{
    meta:
        author = "Odin by Brandefense"
        monograph = "29 / 30"
        classification = "TLP:CLEAR"
        description = "Matches P2PInfect peer-to-peer cloud worm bootstrap and gossip configuration manifests"
    strings:
        $p1 = "X-Peer-Node" ascii
        $p2 = "/p2p/peers" ascii
        $g1 = "60103" ascii
        $g2 = "60140" ascii
        $r1 = "redis_peer_sync" ascii nocase
    condition:
        filesize < 512KB and ($p1 and $p2) and ($g1 or $g2 or $r1)
}
```

```
rule ODIN_EvilGoPhish_Hybrid_AiTM_Config
{
    meta:
        author = "Odin by Brandefense"
        monograph = "29 / 30"
        classification = "TLP:CLEAR"
        description = "Matches EvilGoPhish hybrid GoPhish and reverse-proxy campaign configuration files"
    strings:
        $e1 = "evilgophish" ascii nocase
        $e2 = "X-EvilGoPhish-Key" ascii
        $a1 = "admin_server" ascii nocase
        $a2 = ":3333" ascii
        $px = "evilginx_sub" ascii nocase
    condition:
        filesize < 512KB and ($e1 or $e2) and ($a1 or $a2) and $px
}
```

odin_hookbot_p2pinfect_evilgophish_recon.rules

SURICATA

```
# Odin by Brandefense - Monograph 29 / 30 Suricata 7.x Detection Suite
# Coverage: HookBot, Octopus, Daam/L3MON, P2PInfect, MooBot, EvilGoPhish, ARL/ScopeSentry/reNgine (Watchlist)
# Classification: TLP:CLEAR

alert http $HOME_NET any -> $EXTERNAL_NET [80,443,2223] (msg:"ODIN [R29] HookBot Android Accessibility Overlay
and VNC State Beacon"; flow:established,to_server; http.method; content:"POST"; http.uri; content:"/api/
hookbot/"; fast_pattern; http.request_body; bsize:>24; content:"\"accessibility\""; content:"\"vnc_state\"";
distance:0; classtype:trojan-activity; sid:9529001; rev:1; metadata:created_at 2026_11_19, deployment perimeter,
tactic command_and_control;)

alert http $HOME_NET any -> $EXTERNAL_NET 80 (msg:"ODIN [R29] Octopus C2 Agent Registration Check-In";
flow:established,to_server; http.method; content:"POST"; http.uri; content:"/api/octopus/"; fast_pattern;
http.request_body; bsize:>20; content:"\"agent_uuid\""; classtype:trojan-activity; sid:9529002; rev:1;
metadata:created_at 2026_11_19, deployment perimeter, tactic command_and_control;)

alert http $HOME_NET any -> $EXTERNAL_NET [443,22222,22533] (msg:"ODIN [R29] Daam or L3MON Android Remote
Management Panel Telemetry"; flow:established,to_server; http.method; content:"POST"; http.uri; content:"/
client_login"; fast_pattern; http.request_body; bsize:>18; content:"\"device_model\""; classtype:trojan-activity;
sid:9529003; rev:1; metadata:created_at 2026_11_19, deployment perimeter, tactic command_and_control;)

alert http $HOME_NET any -> $EXTERNAL_NET [6088,8281,9684,60103:60140] (msg:"ODIN [R29] P2PInfect Cloud Worm Peer
Bootstrap Request"; flow:established,to_server; http.uri; content:"/p2p/peers"; fast_pattern; http.header;
bsize:>16; content:"X-Peer-Node:"; classtype:trojan-activity; sid:9529004; rev:1; metadata:created_at 2026_11_19,
deployment perimeter, tactic initial_access;)

alert http $HOME_NET any -> $EXTERNAL_NET 80 (msg:"ODIN [R29] MooBot Edge Worm Stage Fetch";
flow:established,to_server; http.uri; content:"/moobot."; fast_pattern; http.user_agent; bsize:>8;
content:"Wget/"; startswith; classtype:trojan-activity; sid:9529005; rev:1; metadata:created_at 2026_11_19,
deployment perimeter, tactic initial_access;)

alert http $HOME_NET any -> $EXTERNAL_NET [443,3322,3333,3334,10002,33335] (msg:"ODIN [R29] EvilGoPhish Hybrid
AiTM Campaign API Dispatch"; flow:established,to_server; http.method; content:"POST"; http.uri; content:"/api/
campaigns/"; fast_pattern; http.header; bsize:>18; content:"X-EvilGoPhish-Key:"; classtype:trojan-activity; sid:
9529006; rev:1; metadata:created_at 2026_11_19, deployment perimeter, tactic initial_access;)

alert http $HOME_NET any -> $EXTERNAL_NET [443,5003,5004,5080,8443] (msg:"ODIN [R29] POLICY Dual-Use ARL Asset
Reconnaissance Task API"; flow:established,to_server; http.method; content:"POST"; http.uri; content:"/api/
task/"; fast_pattern; http.request_body; bsize:>18; content:"\"task_tag\""; classtype:policy-violation; sid:
9529007; rev:1; metadata:created_at 2026_11_19, deployment perimeter, policy dual_use_watchlist;)

alert http $HOME_NET any -> $EXTERNAL_NET [443,8000,8082,8443,18082] (msg:"ODIN [R29] POLICY Dual-Use ScopeSentry
or reNgine Scan Dispatch"; flow:established,to_server; http.method; content:"POST"; http.uri; content:"/api/scan/
create"; fast_pattern; http.request_body; bsize:>20; content:"\"scan_engine\""; classtype:policy-violation; sid:
9529008; rev:1; metadata:created_at 2026_11_19, deployment perimeter, policy dual_use_watchlist;)
```

sentinel_mobile_overlay_p2pinfect_and_evilgophish_hunt.kql

KQL (SENTINEL / DEFENDER XDR)

```
// Odin by Brandefense - Monograph 29 / 30 Microsoft Sentinel & Defender XDR Hunt
// Correlates confirmed HookBot C2 egress, P2PInfect :601xx worm peers, and EvilGoPhish :3333 activity
// Classification: TLP:CLEAR
// The approved-host exclusion applies only to the port and process branches.
// Contact with a confirmed C2 address always alerts, even from an approved host.

let Lookback = 7d;
let ApprovedReconHosts = dynamic(["ASM-SCANNER-01", "REDTEAM-PHISH-SIM-01", "APPSEC-RECON-01"]);
let ConfirmedMobileC2IPs = dynamic(["152.70.24.49"]);
let MonitoredFleetPorts = dynamic([3333, 3334, 5003, 6088, 8082, 22222, 22533, 60103, 60104, 60136, 60139,
60140]);
let SuspiciousHostProcs = DeviceProcessEvents
    | where Timestamp >= ago(Lookback)
    | where not(DeviceName in~ (ApprovedReconHosts))
    | where ProcessCommandLine has_any ("p2pinfect", "evilgophish", "l3mon", "scopesentry", "rengine", "arl-
worker")
        or (FileName in~ ("redis-server", "sshd", "node", "python3") and ProcessCommandLine has_any ("60103",
"22533", "3333", "slaveof"))
    | project ProcTime = Timestamp, DeviceId, DeviceName, InitiatingProcessFileName, FileName,
ProcessCommandLine;
DeviceNetworkEvents
| where Timestamp >= ago(Lookback)
| extend IsConfirmedC2 = RemoteIP in (ConfirmedMobileC2IPs)
| where IsConfirmedC2
    or (not(DeviceName in~ (ApprovedReconHosts))
        and RemotePort in (MonitoredFleetPorts)
        and InitiatingProcessFileName in~ ("redis-server", "python3", "node", "curl", "wget", "gophish"))
| summarize
    NetFirstSeen = min(Timestamp),
    NetLastSeen = max(Timestamp),
    DistinctRemoteIPs = dcount(RemoteIP),
    DistinctPorts = make_set(RemotePort, 16),
    MatchedC2IPs = make_set_if(RemoteIP, IsConfirmedC2, 16),
    ConnectionCount = count()
    by DeviceId, DeviceName, InitiatingProcessFileName
| join kind=leftouter (SuspiciousHostProcs) on DeviceId, DeviceName
| where array_length(MatchedC2IPs) > 0 or isnotempty(ProcessCommandLine) or DistinctRemoteIPs >= 10
| extend AlertAction = case(
    array_length(MatchedC2IPs) > 0, "BLOCK_CONFIRMED_HOOKBOT_C2",
    ProcessCommandLine has_any ("scopesentry", "rengine", "arl-worker"), "REVIEW_UNAPPROVED_DUAL_USE_ASM",
    "HUNT_P2PINFECT_WORM_OR_EVILGOPHISH_NODE"
)
| project NetFirstSeen, NetLastSeen, DeviceName, AlertAction, MatchedC2IPs, DistinctRemoteIPs, DistinctPorts,
InitiatingProcessFileName, ProcessCommandLine, ConnectionCount
| sort by ConnectionCount desc
```

sigma_p2pinfect_redis_ssh_worm_and_evilgophish.yml

SIGMA

```
title: P2PInfect Cloud Worm Redis or SSH Peer Spawning
id: 4c8e2a91-7b31-49d4-9e12-202611192901
status: experimental
description: Detects Linux Redis or SSH daemon child processes associated with P2PInfect peer-to-peer worm
propagation across ports 60103 through 60140 and 6088.
author: Odin by Brandefense
date: 2026/11/19
tags:
  - attack.initial_access
  - attack.command_and_control
logsource:
  category: process_creation
  product: linux
detection:
  selection_parent:
    ParentImage|endswith:
```

```
      - '/redis-server'
      - '/sshd'
  selection_worm:
    CommandLine|contains:
      - '60103'
      - '60104'
      - '60140'
      - '6088'
      - 'p2pinfect'
      - 'slaveof'
  filter_approved_db:
    ComputerName:
      - 'REDIS-REPLICA-PROD-01'
  condition: selection_parent and selection_worm and not filter_approved_db
falsepositives:
  - Sanctioned Redis replication clusters using custom high ports listed in asset inventory
level: high
---
title: Unapproved EvilGoPhish Hybrid AiTM Campaign Server Execution
id: 7f291c40-6e82-4a15-8b33-202611192902
status: experimental
description: Identifies unapproved execution of EvilGoPhish hybrid GoPhish and Evilginx reverse-proxy phishing
infrastructure binding port 3333 or 3334.
author: Odin by Brandefense
date: 2026/11/19
tags:
  - attack.initial_access
  - attack.command_and_control
logsource:
  category: process_creation
  product: linux
detection:
  selection_cli:
    CommandLine|contains:
      - 'evilgophish'
      - 'evilginx'
      - 'gophish_admin'
      - ':3333'
  filter_redteam:
    ComputerName:
      - 'REDTEAM-PHISH-SIM-01'
  condition: selection_cli and not filter_redteam
falsepositives:
  - Authorized security-awareness phishing simulation hosts registered with security operations
level: high
---
title: Unapproved Dual-Use ARL ScopeSentry or reNgine Recon Worker Launch
id: 9a540b18-2c77-4d89-b160-202611192903
status: experimental
description: Audits unapproved launches of dual-use ARL (:5003), ScopeSentry (:8082), or reNgine (:8082) attack-
surface reconnaissance workers for policy review.
author: Odin by Brandefense
date: 2026/11/19
tags:
  - attack.reconnaissance
logsource:
  category: process_creation
  product: linux
detection:
  selection_asm:
    CommandLine|re: '(arl[-_](worker|web)|scopesentry.*8082|rengine[-_]celery)'
  filter_asm_hosts:
    ComputerName:
      - 'ASM-SCANNER-01'
      - 'APPSEC-RECON-01'
  condition: selection_asm and not filter_asm_hosts
falsepositives:
  - Sanctioned external attack surface management scanners; route to dual-use watchlist review
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/mobile-overlay-c2s-cloud-worms-and-recon-phishing-fleets/
Short Read · brandefense.io/odin-research/blog/banking-trojan-panels-cloud-worms-and-phishing-kits/
Defense Kit · brandefense.io/odin-research/defense-kit/