

Inside ImmoRan Static and Vehere Sandbox Detonation of the ImmoRan Go Ransomware Builder

A Technical Threat Research Report

Family

NightSpire ("NSPIRE"); internal builder project ImmoRan

Suspected lineage

Probable rebrand of the Rbfs RaaS operation

Platform

Windows PE32+ x64, console subsystem (pure Go, CGO disabled)

IOCs

2c3f8889c9f5d765632ad91b4cc2a852c54845e143dbd2b68359b6f9cd8f99 (Sample A)

c5f526cc62688cf34c49d098dab81e24e4294f832ada57433ef505d5ac6da8f3 (Sample B)

Analysis method

Static binary teardown plus dynamic detonation in Vehere Sandbox.

Publication date

August 2026

Author

Kumar Pranjal Shukla

Co-Author

Subhamoy Som

Executive Summary

This report presents static analysis and Vehere Sandbox detonation results for two related NightSpire ransomware builds.

The analysis also corrects several claims that have circulated in open-source reporting on this family. These builds use per-build **RSA-4096 public keys** rather than a shared RSA-2048 key, perform **bulk encryption with AES** rather than ChaCha20/HKDF, and embed per-victim negotiation identifiers that had previously been characterized as anti-virtual-machine checks.

Build metadata further recovers the operator's internal project name, ImmoRan, together with a durable folder typo ("Immotal") introduced in the second build. At the time of writing, neither string was indexed on major public code-hosting platforms, making them practical hunting pivots that should survive future rebranding of the public NightSpire name.

Key Judgment

NightSpire is assessed with moderate confidence to be a rebrand of the Rbfs ransomware-as-a-service operation, associated with operators tracked as xdragon128 and cuteliyuan. Open-source reporting from mid-2026 indicates the group was largely dormant following an active operational period through October 2025, though the builder toolchain analyzed here continued to receive updates into 2026.

Key Findings

Bulk encryption is AES wrapped under per-build RSA-4096 keys. No ChaCha20 or HKDF routines are present in either sample.

Strings such as NSPIREVMKR9GEWW and NSPIREUMK44ACC9 are per-build negotiation UUIDs referenced in the ransom-note text, not virtual-machine or sandbox self-checks.

Comparison of the two builds shows maturation from an informal, target-specific extortion note (v1) to a standardized double-extortion note with a structured leak timeline (v2).

Threat Actor Profile

Open-source reporting associates NightSpire with a probable rebrand of the Rbfs operation. The table below summarizes the attribution picture used for this report; claims outside the two analyzed encryptors are sourced from third-party tracking.

Attribute	Assessement
Public family name	NightSpire
Internal builder project	ImmoRan ("Immotal" typo in v2 build path)
Suspected prior identity	Rbfs RaaS
Associated operators	xdragon128, cuteliyuan (previously linked to Paranodeus / CyberVolk / DarkAssault activity)
Extortion model	Double extortion — encryption plus data-leak threat via Tor leak site
Primary initial access vector	CVE-2024-55591 (FortiOS / FortiProxy authentication bypass); RDP brute-forcing
Observed post-exploitation tooling	Chrome Remote Desktop, AnyDesk, Everything, Advanced IP Scanner, 7-Zip, MEGA sync/MEGACmd (Tor-proxied), WinSCP, Mimikatz, PsExec, PowerShell/WMI
Operational tempo	Active campaign through approximately October 2025; assessed as largely dormant as of mid-2026 per third-party reporting
Estimated victim count	Approximately 259 disclosed victims as of May 2026 (third-party tracking)

Table 1 — Threat-actor profile from open-source reporting

Communication Infrastructure

Victim contact and leak infrastructure is Tor-centric, with email and qTox as secondary channels. Values recovered from the analyzed samples are marked accordingly; additional addresses are included from third-party reporting.

Channel	Value	Status
Primary leak-site onion	a2lyiaq4n74tlgz4fk3ft4akolapfrzk772dk24iq32cnjsmzpanqd[.]onion	Previously documented
Contact-page onion	nspireyzmvapgiwgtuoznlafqvlyz7ey6himtgn5bdvdcowfyto3yryd[.]onion/contact[.]php	Embedded in Sample B
Active leak-site onion	nspirebcv4sy3yydtaercuut34hwc4fsxqqv4b4ye4xmo6qp3vxhulqd[.]onion	Corroborated
Sibling-variant onion	nspire7lugml7ybqyjaaxtsgrs4qn3fcon3lrjbih6wamttvdm5ke4qd[.]onion	Related family sample
Additional onion	nspiremkiq44[.]onion	Third-party reporting
Contact e-mail	nightspireteam[.]receiver@onionmail[.]org	Embedded in both samples
Additional e-mails	night[.]spire[.]team@gmail[.]com; night[.]spire[.]team@onionmail[.]org; night[.]spire[.]team@proton[.]me; nightspire[.]team2026@onionmail[.]org	Third-party reporting
qTox ID (v1 sample)	3B61CFD6E12D789A439816E1DE08CFDA58D76EBOB26585AA34CDA617C41D5943CDD15DB0B7E6	Previously published
qTox ID (v2 sample)	8D663FD10BF662930F4C076CBF95FACFCC4ABD8F1A5E328DE75DOB0237A74E1AE1EOC5C37E7F	Newly recovered from this sample

Table 2 — Victims contact and leak infrastructure

Samples Under Analysis

The following two PE32+ x64 console binaries were selected for side-by-side static comparison. Both are pure Go (CGO disabled) with kernel32-only imports.

Attribute	Sample A [v1]	Sample B [v2]
SHA-256	2c3f8889c9f5d765632ad91b4cc2a852c54845e143dbd2b68359b6f9cdfd8f99	c5f526cc62688cf34c49d098dab81e24e4294f832ada57433ef505d5ac6da8f3
Format	PE32+ x64, console subsystem	PE32+ x64, console subsystem
Go compiler version	go1.24.1	go1.24.11
Image base	0x400000	0x400000
Section count	8 (.text .rdata .data .pdata .xdata .idata .reloc .symtab)	8 (identical layout)
Imports	kernel32.dll only (46 functions)	kernel32.dll only
.text entropy	~6.25	~6.25

Table 3 — Samples under analysis

Public Availability and Third-Party Corroboration

Both samples are independently indexed by third-party platforms. MalwareBazaar lists both SHA-256 hashes under NightSpire. AttackIQ published an emulation template against Sample B that independently confirms the UUID is embedded at build time; its AES-256-CBC/RSA-2048 description is family-generic and does not override the RSA-4096 recovery from these two binaries. Huntress has documented a related enc.exe / [nspire_msg].txt variant. Endpoint vendors detect the family under names including SonicWall GAV: NightSpire.RSM and AhnLab .C5769860 / .C5775165.

Build Provenance: The ImmoRan / Immotal Trail

Go build-info metadata. Both samples embed the operator's local development layout/footprint:

Sample A:

C:/Users/Administrator/Documents/Dev_Pack/Encryp-
ter/New folder/ImmoRan/main.go

Sample B:

C:/Users/Administrator/Documents/Dev_Tools/Encry-
pter/Immotal Ransomware(v2)/ImmoRan(v2)/main.go

```
$ go version -m sample_a_2c3f8889.exe
sample_a_2c3f8889.exe: go1.24.1
@path@command-line-arguments
@build@-buildmode=exe
@build@-compiler=gc
@build@-ldflags="-s -w"
@build@CGO_ENABLED=0
@build@GOARCH=amd64
@build@GOOS=windows
@build@GOAMD64=v1

$ go version -m sample_b_c5f526cc.exe
sample_b_c5f526cc.exe: go1.24.11
@path@command-line-arguments
@build@-buildmode=exe
@build@-compiler=gc
@build@-ldflags="-s -w"
@build@CGO_ENABLED=0
@build@GOARCH=amd64
@build@GOOS=windows
@build@GOAMD64=v1
```

Figure 1 — Go build-info comparison (go1.24.1 vs go1.24.11)

Victim Negotiation Identifiers

Embedded alphanumeric strings in both samples have previously been described as virtual-machine or sandbox markers. Offset-level review of the embedded ransom-note text shows they are instead per-build negotiation identifiers presented to the victim as an authentication token for the contact portal.

Sample	Identifier	Offset	Context
A	NSPIREVMKR9GEWW	0x18c20e	"Send this UUID so make sure, it is you"
B	NSPIREUMK44ACC9	0x18d557/0x18dc8d	"First say UUID"

Table 4 — Victim negotiation UUID identifiers

```
# Sample A - offset 0x18c20e
0018c200 ... Send this UUID "NSPIREVMKR9GEWW" so make sure, it is you

# Sample B - offset 0x18d557 / 0x18dc8d
0018d550 ... First say UUID : "NSPIREUMK44ACC9"
```

Figure 2 — UUID strings sit inside the ransom-note text, not in an anti-VM routine.

Cryptographic Architecture

Package and Primitive Inventory

Component	Observed
crypto/rsa	Present (x32 references)
crypto/aes/crypto/cipher	Present (x12 / x12 references)
crypto/x509	Present (x19 references)
ed25519	Present (x24 references) — used for signature generation, not bulk encryption
sha256	Present (x40 references)
crypto/internal/fips140	Present (x736 references) - standard Go 1.24 toolchain artifact, not custom hardening
ChaCha20	Absent
HKDF	Absent

Table 5 — Cryptographic package and primitive inventory

Recovered Function Surface

The following function names were recovered directly from pcIntab, which remains populated even though the COFF symbol table was stripped at compile time.

- main.EncryptRSA
- main.generateSalt
- main.changeFileName
- main.bytesToPublicKey
- main.checkPossibility
- main.startEncrypting
- main.GenerateRandomKey
- main.EncryptFile
- main.MakeReadMeFile
- main.generateSignValue (ed25519)
- main.getEncryptionType
- main.AddedSpecifier
- main.GenerateRandomNonce
- main.EncryptDir
- main.writeToTail
- main.encryptMethod

```
main.AddedSpecifier
main.EncryptDir
main.EncryptFile
main.EncryptRSA
main.GenerateRandomKey
main.GenerateRandomNonce
main.MakeReadMeFile
main.bytesToPublicKey
main.changeFileName
main.checkPossibility
main.encryptMethod
main.generateSalt
main.generateSignValue
main.getEncryptionType
main.startEncrypting
main.writeToTail
main.main
```

Figure 3 — Function names from samples

Embedded Public Keys

Sample	SPKI SHA-256 Fingerprint
A	ca343308d30c686d77acb46384b44987b88d167d10d3f43edd2e6e0d8e02b5a6
B	d0332ea23d19e67d4a28f8d3ec02e8b37cf3a98d4738d7a75fc33d67d930008a

Table 6— Per-sample RSA-4096 SPKI fingerprints

Both recovered keys are RSA-4096 with public exponent 65537, encoded as a 550-byte DER SubjectPublicKeyInfo structure, and are distinct between builds—indicating per-build keypair generation rather than a single key reused across victims. Sample B's ransom note self-describes the scheme as AES-256 / RSA-4096, consistent with the static recovery.

```
# Embedded RSA public keys (DER SPKI, e=65537)

Sample A (0x18b8bc)
  modulus      : 4096-bit RSA, e=65537
  SPKI sha256  : ca343308d30c686d77acb46384b44987b88d167d10d3f43edd2e6e0d8e02b5a6

Sample B (0x18cf71)
  modulus      : 4096-bit RSA, e=65537
  SPKI sha256  : d0332ea23d19e67d4a28f8d3ec02e8b37cf3a98d4738d7a75fc33d67d930008a

# Distinct per-build keys. Matches the note claim AES-256 / RSA-4096.
# Updates earlier public reporting that described RSA-2048.
```

Figure 4 — Distinct RSA-4096 SPKI fingerprints for Sample A and Sample B.

Correction to Public Reporting

Static evidence from these two builds does not support the RSA-2048 / ChaCha20 / HKDF description that has circulated in some open-source coverage of this family. These builds use AES for bulk file encryption with the resulting key and nonce wrapped under a per-build RSA-4096 public key. Other builds of this family may differ; this finding is scoped to the two samples analyzed.

Binary and Encryption Workflow

The PE layout and the high-level encryption / note-drop workflows below are reconstructed from the recovered function surface and static string inventory.

PE Section Layout

Both samples share the same eight-section layout. Entropy and import characteristics are consistent with an unpacked, statically linked Go binary.

Section	Contents
.text	Compiled Go code (entropy ~6.25); no packing indicators
.rdata	Go pclntab, string constants, ransom-note text, embedded RSA public key
.data/.pdata/.xdata	Standard Go runtime data structures
.idata	Import table — kernel32.dll only (46 functions)
.reloc/.symtab	Relocation table; residual Go symbol-table section header

Table 7 - Shared PE section layout

File Encryption Workflow

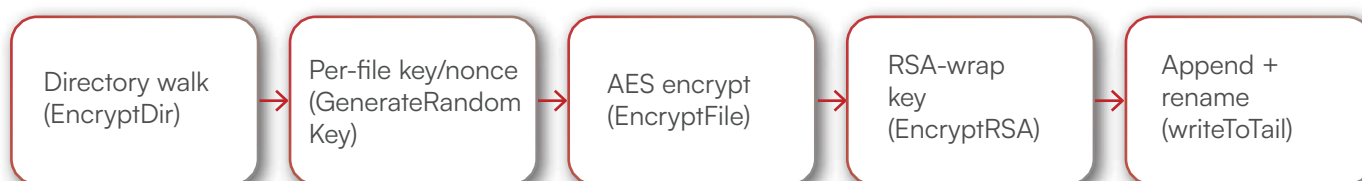


Table 8 — Reconstructed file-encryption workflow

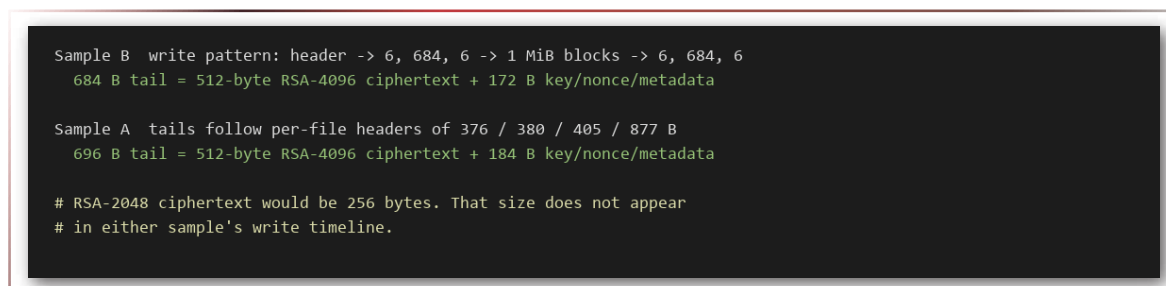


Figure 5 — Per-file tail sizes (696 B / 684 B) match RSA-4096, not RSA-2048

Ransom Note Deployment Workflow

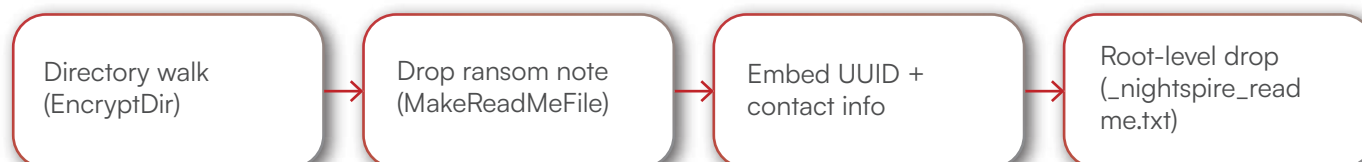


Table 9 — Reconstructed file-encryption workflow

Extortion Model Evolution

Side-by-side comparison of ransom-note content and deployment behavior shows a clear progression from a targeted, informal extortion approach to a standardized double-extortion model.

Characteristic	Sample A [v1]	Sample B [v2]
Note greeting	“Hi, MasterBaker Guys!” (target-specific)	“NightSpire Encryption Notice” (standardized)
Note deployment	Dropped across 28 distinct directories	Single drop at drive root: C:_nightspire_readme.txt
Encryption claim in note	Not specified	“AES-256 / RSA-4096”
Payment structure	Unstructured, no deadline	Staged demands with a leak-schedule timeline
Leak schedule	None	Day 10 — blog notice; Day 14 — full data dump
Data exfiltration menu	Not present	Structured categories: PII, credentials, financial data, source code
Upsell	None	Additional \$50,000 “security audit” offer

Table 10 — Extortion-model evolution between Sample A and Sample B

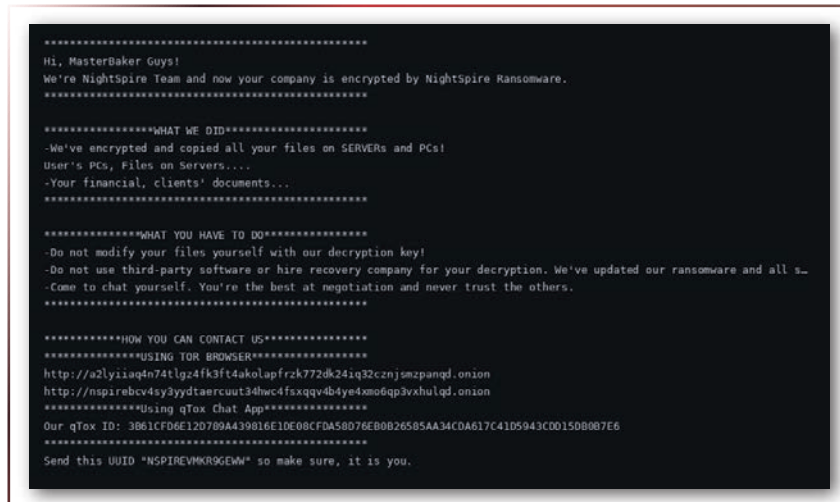


Figure 6 — Sample A ransom note (target-specific greeting; UUID NSPIREVMKR9GEWW).

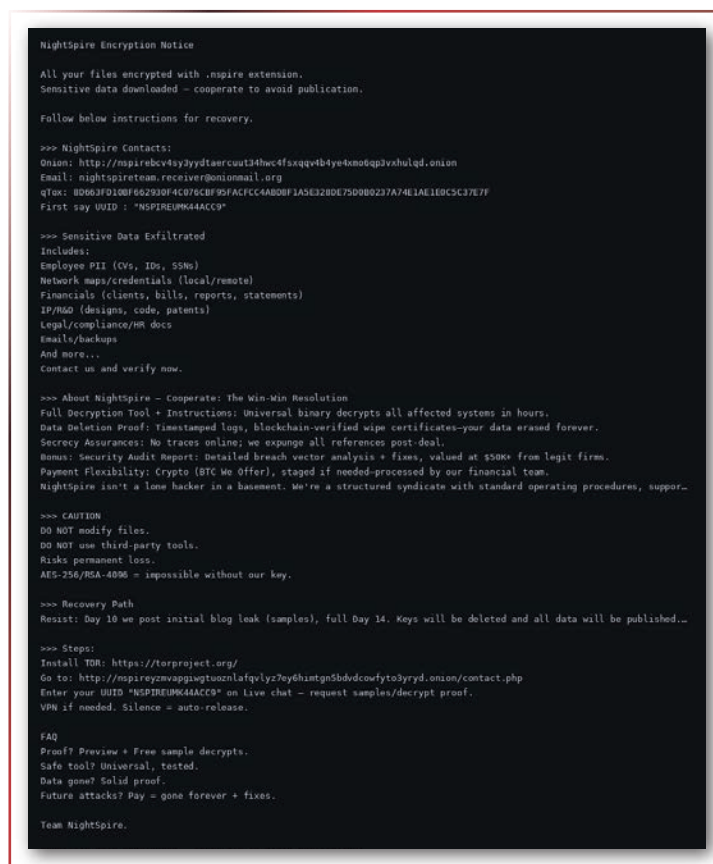


Figure 7 — Sample B ransom note (AES-256/RSA-4096; Day 10 / Day 14 leak clock).

Anti-Analysis Timing Behavior

Both samples implement a thread suspend/resume loop consistent with timing-based sandbox evasion. The intensity of this behavior increases substantially between builds. This was detected by **PRIM_THREAD_SUSPEND_RESUME_LOOP**, *Vehere's exclusive Sandbox signature*.

Sample	Suspend/Resume Pairs	Threads Involved
A	31	4
B	161	5

Table 11 — Thread suspend/resume timing behavior

Dynamic Analysis

Both samples were detonated to validate the static findings against runtime behavior. Detonation was performed without network egress; the observations below reflect host-level behavior only.

Sample A – Detonation Summary

Live detonation metadata for Sample A is summarized below.

Sample A – Vehere Detections

Behavioral signatures reported for Sample A:

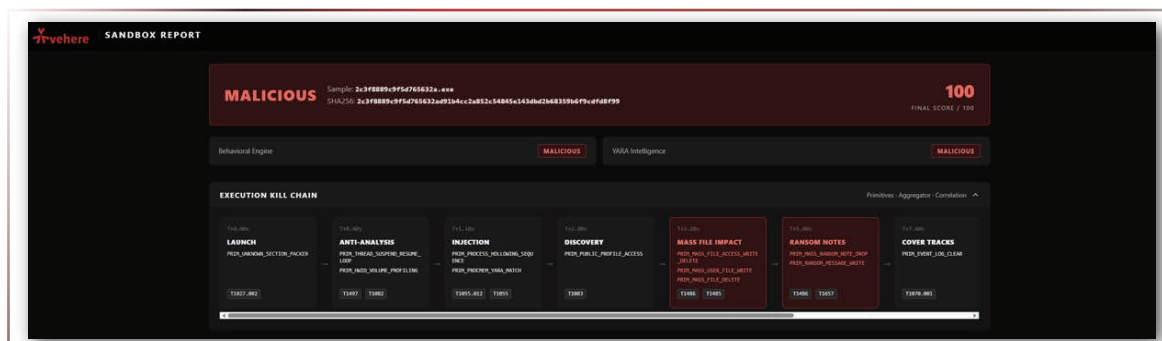


Figure 8 — Sample A sandbox report with execution kill chain (final score 100.0 / 100, malicious verdict)

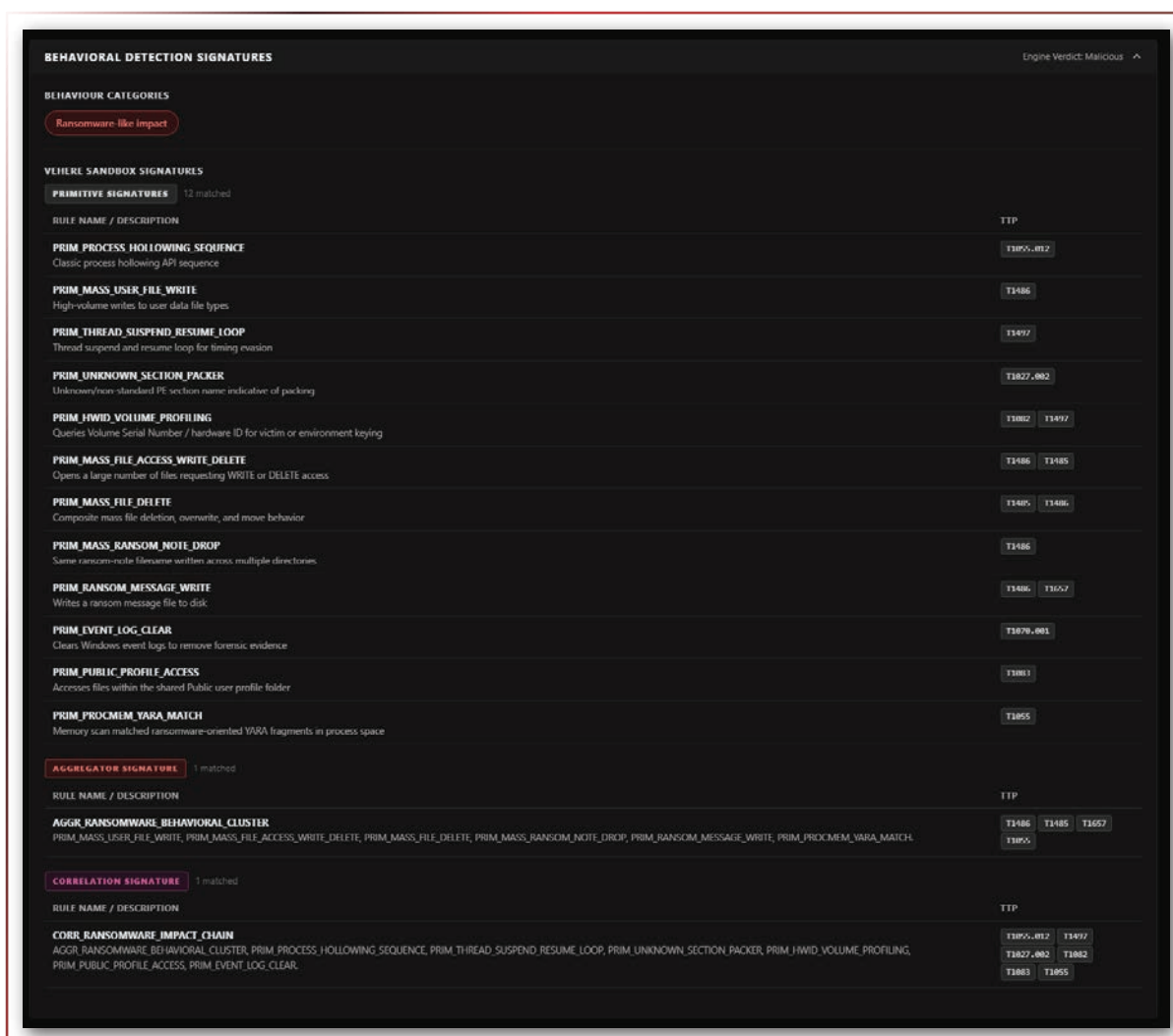


Figure 9 — Sample A behavioral signatures (malicious)

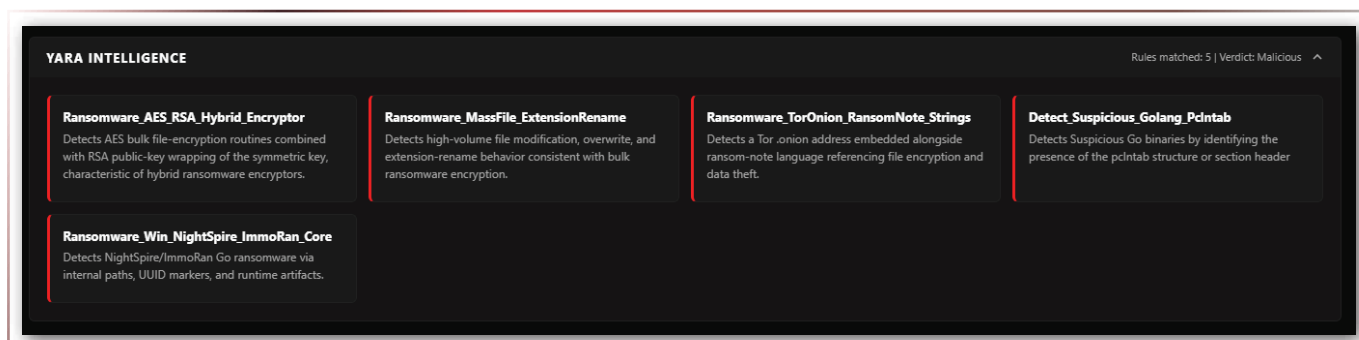


Figure 10 — Sample A YARA intelligence (5 rules matched, including Ransomware_Win_NightSpire_ImmoRan_Core)

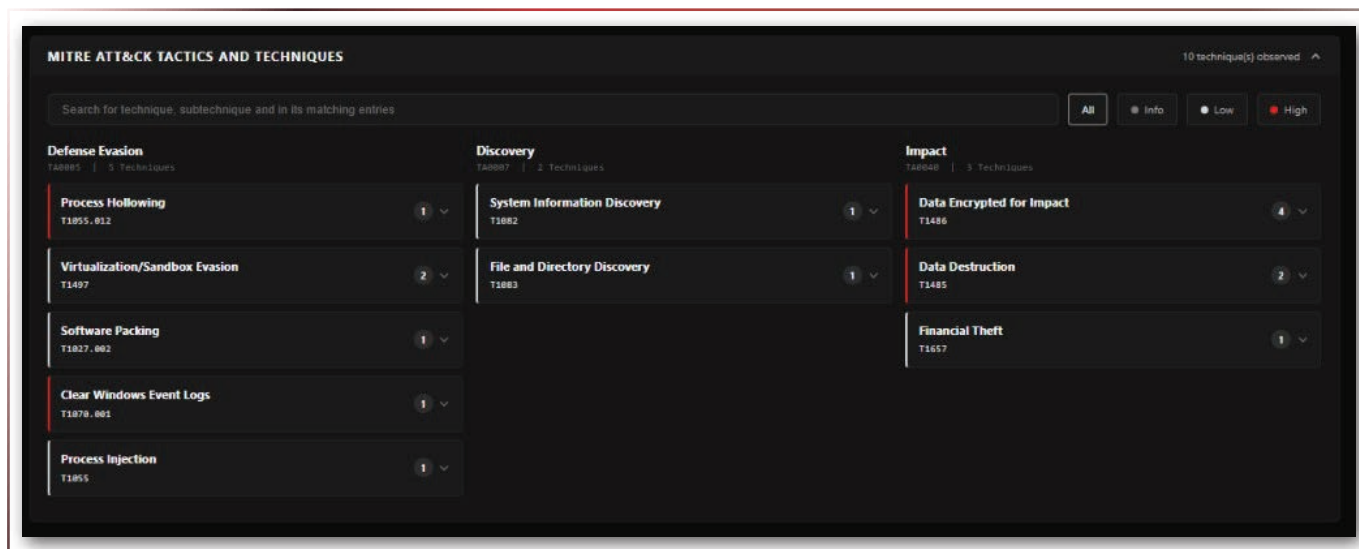


Figure 11 — Sample A MITRE ATT&CK tactics and techniques (10 techniques observed)

Process hollowing (T1055.012) was reported by the sandbox engine for Sample A and was not evident from the static encryptor surface reconstructed in Sections 6—7. This technique should be treated as engine-detected pending further static confirmation of an injection path.

Sample B — Detonation Summary

The runtime alias enc.exe recovered during detonation matches third-party reporting that attributed an enc.exe artifact to this family, confirming it as an observed filename for this sample rather than merely an unconfirmed sibling.

Sample B — YARA Intelligence

Sample B matched the same YARA intelligence rule set as Sample A, including Ransomware_Win_NightSpire_ImmoRan_Core (Figure 10).

Cross-Sample Behavioral Comparison

Event-log clearing and broader directory traversal (including the public profile) appear in the Sample A run but not in Sample B. That difference is consistent with the static finding that v1 scattered its ransom note across many directories, while v2 drops a single note at the drive root.

Dynamically Confirmed ATT&CK Techniques

Vehere Sandbox Signature	MITRE ATT&CK	Type
PRIM_PROCESS_HOLLOWING_SEQUENCE	T1055.012	Primitive
PRIM_MASS_USER_FILE_WRITE	T1486	Primitive
PRIM_THREAD_SUSPEND_RESUME_LOOP	T1497	Primitive
PRIM_UNKNOWN_SECTION_PACKER	T1027.002	Primitive
PRIM_HWID_VOLUME_PROFILING	T1082, T1497	Primitive
PRIM_MASS_FILE_ACCESS_WRITE_DELETE	T1486, T1485	Primitive
PRIM_MASS_FILE_DELETE	T1485, T1486	Primitive
PRIM_MASS_RANSOM_NOTE_DROP	T1486	Primitive
PRIM_RANSOM_MESSAGE_WRITE	T1486, T1657	Primitive
PRIM_EVENT_LOG_CLEAR	T1070.001	Primitive
PRIM_PUBLIC_PROFILE_ACCESS	T1083	Primitive
PRIM_PROCMEM_YARA_MATCH	T1055	Primitive
AGGR_RANSOMWARE_BEHAVIORAL_CLUSTER	T1486, T1485, T1657, T1055	Aggregator
CORR_RANSOMWARE_IMPACT_CHAIN	T1055.012, T1497, T1027.002, T1082, T1083, T1055	Correlation

Table 12 — Sample A behavioral signature rules and ATT&CK mapping

Technique	Name	Evidence Basis
T1486	Data Encrypted for Impact	.nspire renames, bulk ciphertext writes, RSA/AES surface, ransom-note drop
T1497	Virtualization/Sandbox Evasion	Thread suspend/resume timing loop and associated kernel32 imports
T1657	Financial Theft	Ransom note plus Tor-based payment and contact infrastructure
T1055.012	Process Hollowing	Sandbox-reported process-hollowing API sequence (Sample A)
T1027.002	Software Packing	Non-standard / unusual PE section indicators reported at launch
T1070.001	Clear Windows Event Logs	Event-log clear behavior observed in the Sample A detonation
T1485	Data Destruction	Mass file delete / overwrite patterns in behavioral signatures
T1082	System Information Discovery	HWID / volume serial profiling during execution
T1083	File and Directory Discovery	Public-profile and directory traversal during encryption workflow

Table 13 — Core ATT&CK techniques with evidence basis

Indicators of Compromise (IOCs)

The indicators below are scoped to the two builds analyzed in this report unless otherwise noted.

File Hashes

SHA-256 (Sample A): 2c3f8889c9f5d765632ad91b4cc2a852c54845e143dbd2b68359b6f9cddf8f99
SHA-256 (Sample B): c5f526cc62688cf34c49d098dab81e24e4294f832ada57433ef505d5ac6da8f3

Ransom Artifacts

- *_nightspire_readme.txt
- *.nspire (encrypted-file extension)
- C:\[NSPIRE_MSG].txt (related family variant, per third-party reporting)
- enc.exe (executable name observed as a Sample B runtime alias; also reported for a related family variant)

Victim Negotiation UUIDs

- NSPIREVMKR9GEWW
- NSPIREUMK44ACC9

Build-Path Pivots

- ImmoRan
- Immortal Ransomware
- Dev_Pack\Encrypter
- Dev_Tools\Encrypter

RSA-4096 SPKI Fingerprints (SHA-256)

Sample A:

ca343308d30c686d77acb463
84b44987b88d167d10d3f43ed
d2e6e0d8e02b5a6

Sample B:

d0332ea23d19e67d4a28f8d3e
c02e8b37cf3a98d4738d7a75fc
33d67d930008a

Network / Contact Infrastructure

Type	Value
Onion (leak site)	a2lyiaq4n74tlgz4fk3ft4akolapfrzk772dk24iq32cznjmzpanqd[.]onion
Onion (contact)	nspireyzmvapgiwgtuoznlafqvlyz7ey6himtgn5bdvdcowfyto3yryd[.]onion
Onion (active leak site)	nspirebcv4sy3yydtaercuut34hwc4fsxqqv4b4ye4xmo6qp3vxhulqd[.]onion
Onion (sibling variant)	nspire7lugml7ybqjaaxtsgrs4qn3fcon3lrjbih6wamttvdm5ke4qd[.]onion
Onion (additional)	nspiremkiq44[...][.]onion
E-mail	Nightspireteam[.]receiver@onionmail[.]org, night[.]spire[.]team@gmail[.]com night[.]spire[.]team@onionmail[.]org, night[.]spire[.]team@proton[.]me nightspire[.]team2026@onionmail[.]org
qTox (v1, previously published)	3B61CFD6E12D789A439816E1DE08CFDA58D76EB0B26585AA34CDA617 C41D5943CDD15DB0B7E6
qTox (v2, newly recovered)	8D663FD10BF662930F4C076CBF95FACFCC4ABD8F1A5E328DE75D0B0 237A74E1AE1EOC5C37E7F

Table 14 — Network and contact infrastructure indicators

Vendor Signature Names

- SonicWall: GAV: NightSpire.RSM
- AhnLab: .C5769860 / .C5775165

Assessment

NightSpire's builder toolchain reflects a small, actively maintained development effort rather than a large engineering organization: unversioned go build invocations, a persistent internal project name across builds, and increasing anti-analysis sophistication between the two samples. Distinctive pivots from this work include the ImmoRan / Immotal build paths, both negotiation UUID markers, and per-build RSA-4096/AES encryption—correcting prior RSA-2048/ChaCha20/HKDF descriptions applied to this family in open-source reporting.

Recommendations

- Hunt on the ImmoRan / Immotal build-path fragments and the two negotiation UUID strings. At present they provide a low false-positive detection surface across endpoint and file-scanning pipelines.
- Prioritise patching or compensating controls for CVE-2024-55591 on internet-facing FortiOS / FortiProxy appliances and monitor for anomalous RDP authentication attempts given this group's documented initial-access preferences.
- Treat the documented onion domains as a related infrastructure set for blocking purposes, recognizing that individual domains are expected to rotate.

Appendix A: Related Family Variant

Third-party reporting documents a related sample using the executable name enc.exe and the ransom-note filename C:[NSPIRE_MSG].txt, along with an additional onion domain (nspire7lugml7ybqyjaaxtsgrs4qn3fcon3lrjbih6wamttvdm5ke4qd[.onion]). Overlapping ATT&CK techniques remain T1486, T1497, and T1657 after exclusion of environment-induced detections. This variant was not independently obtained or statically analyzed for this report.

References

The following external sources support attribution, infrastructure, sample corroboration, and vulnerability context cited in this report. Vehere Sandbox detonation results and static analysis of Samples A and B are original to this work.

1. **S-RM Intelligence.** Ransomware in focus: Meet NightSpire. <https://www.s-rminform.com/latest-thinking/ransomware-in-focus-meet-nightspire> - Threat-actor attribution, Rbfs lineage, and operator handles (xdragon128, cuteliyuan).
2. **Halcyon.** NightSpire threat group profile. <https://www.halcyon.ai/threat-group/nightspire> - Group profile, initial-access patterns, and operational tempo assessment.
3. **Huntress.** Decoding NightSpire: Ransomware IOCs Aren't Set in Stone. <https://www.huntress.com/blog/nightspire-ransomware> - Related enc.exe encryptor and [nspire_msg].txt ransom-note variant observations.
4. **AttackIQ.** Emulating the Persuasive NightSpire Ransomware. <https://www.attackiq.com/2026/04/14/nightspire-ransomware/> - Independent emulation of Sample B (SHA-256 c5f526cc62688cf34c49d098dab81e24e4294f832ada57433ef505d5ac6da8f3) and embedded negotiation UUID.
5. **MalwareBazaar.** NightSpire malware signature listing. <https://bazaar.abuse.ch/browse/signature/NightSpire/> - Public indexing of NightSpire-tagged samples, including both analysed SHA-256 hashes.
6. **Fortinet PSIRT.** FG-IR-24-535: FortiOS and FortiProxy authentication bypass (CVE-2024-55591). <https://fortiguard.fortinet.com/psirt/FG-IR-24-535> - Vendor advisory for the FortiOS / FortiProxy authentication bypass cited in Recommendations.
7. **CISA.** Known Exploited Vulnerabilities Catalog: CVE-2024-55591. https://www.cisa.gov/known-exploited-vulnerabilities-catalog?field_cve=CVE-2024-55591 - U.S. government tracking of in-the-wild exploitation for CVE-2024-55591.
8. **Broadcom Symantec.** NightSpire Ransomware protection bulletin. <https://www.broadcom.com/support/security-center/protection-bulletin/nightspire-ransomware> - Vendor detection naming context (including third-party AV labels for the family).
9. **Ransomware.live.** NightSpire group tracking. <https://www.ransomware.live/group/nightspire> - Public victim-count and leak-site tracking referenced in the threat-actor profile table.