MALWARE ANALYSIS REPORT

AgentTesla sample 2e5ff37c3fbb (corpus)

SHA-256 2e5ff37c3fbba7de… · corpus track · x86

SEVERITY	CONFIDENCE	TLP
MEDIUM	HIGH	TLP:AMBER

Report ID	`MCA-1777844690`
Generated	2026-05-03 21:44:50 UTC
Analyst	MCA Pipeline
Pipeline	`0.1.0`
Malware Family	**AgentTesla**

KEY JUDGMENTS

MEDIUM Family attribution: AgentTesla (Spyware, x86). Severity score 0.34, difficulty 4.63.

MEDIUM 5 MITRE techniques pre-attributed in pe_static_v1; top: T1555.003, T1622, T1057, T1055.001

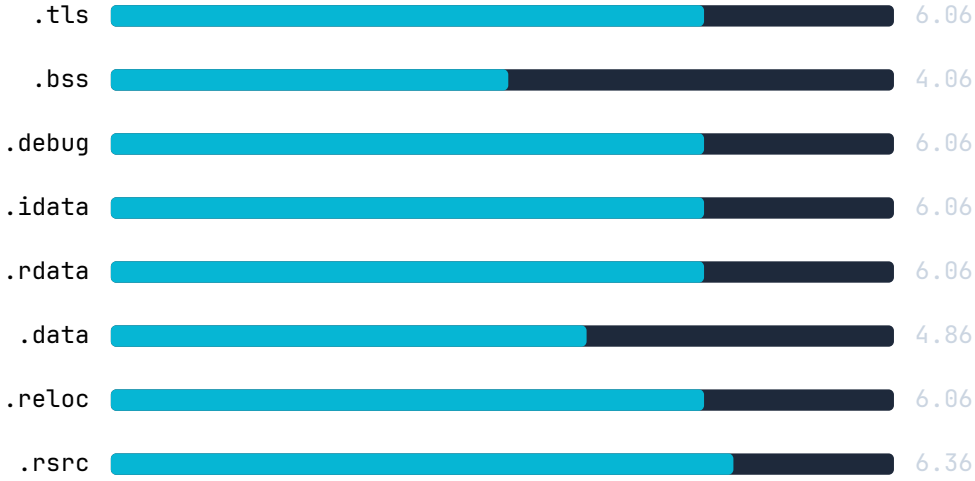
Executive Summary

Static analysis of AgentTesla (Spyware). Architecture x86; entropy 6.06; 8 sections; 17 suspicious imports. 5 MITRE ATT&CK techniques attributed. Severity: MEDIUM.

Submitted Samples

FILENAME	SHA-256	SIZE	TYPE
AgentTesla_2e5ff37c3fbb.bin	2e5ff37c3fbba7decb57eea3c94e70e2d383e8fe0fb54aee3ae16a4aea3f73e7	18.8 KB	PE32

Section Entropy



Entropy ≥ 7.0 typically indicates packing or encryption.

Static Analysis

Corpus profile (pre-ingested static facts):

```
family: AgentTesla
type: Spyware
threat: low
arch: x86
packed: False  packer: none
entropy: 6.0573
sections: .tls, .bss, .debug, .idata, .rdata, .data, .reloc, .rsrc
suspicious_imports: 17
mitre: T1555.003, T1622, T1057, T1055.001, T1056.001
anti_debug: False  anti_vm: False  injection: False
net: True  file_mod: False  reg_mod: False
code:
// AgentTesla - synthetic analysis
#include <windows.h>
void execute() {
  NtQueryInformationProcess(NULL, NULL, NULL, NULL);
  LoadLibrary(NULL, NULL, NULL, NULL);
  IsDebuggerPresent(NULL, NULL, NULL, NULL);
  ZwSetInformationThread(NULL, NULL, NULL, NULL);
  WSAStartup(NULL, NULL, NULL, NULL);
}
```

Evasion & Anti-Analysis

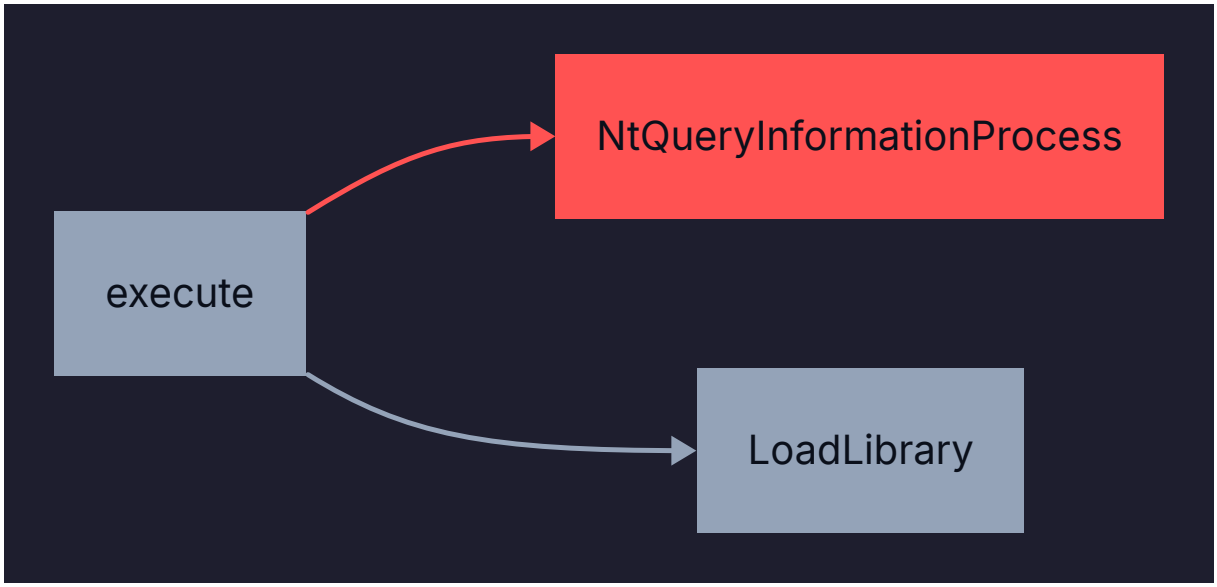
Anti-analysis imports: NtQueryInformationProcess. The sample probes for a debugger or sandbox before executing its main payload.

Decompiled Excerpt

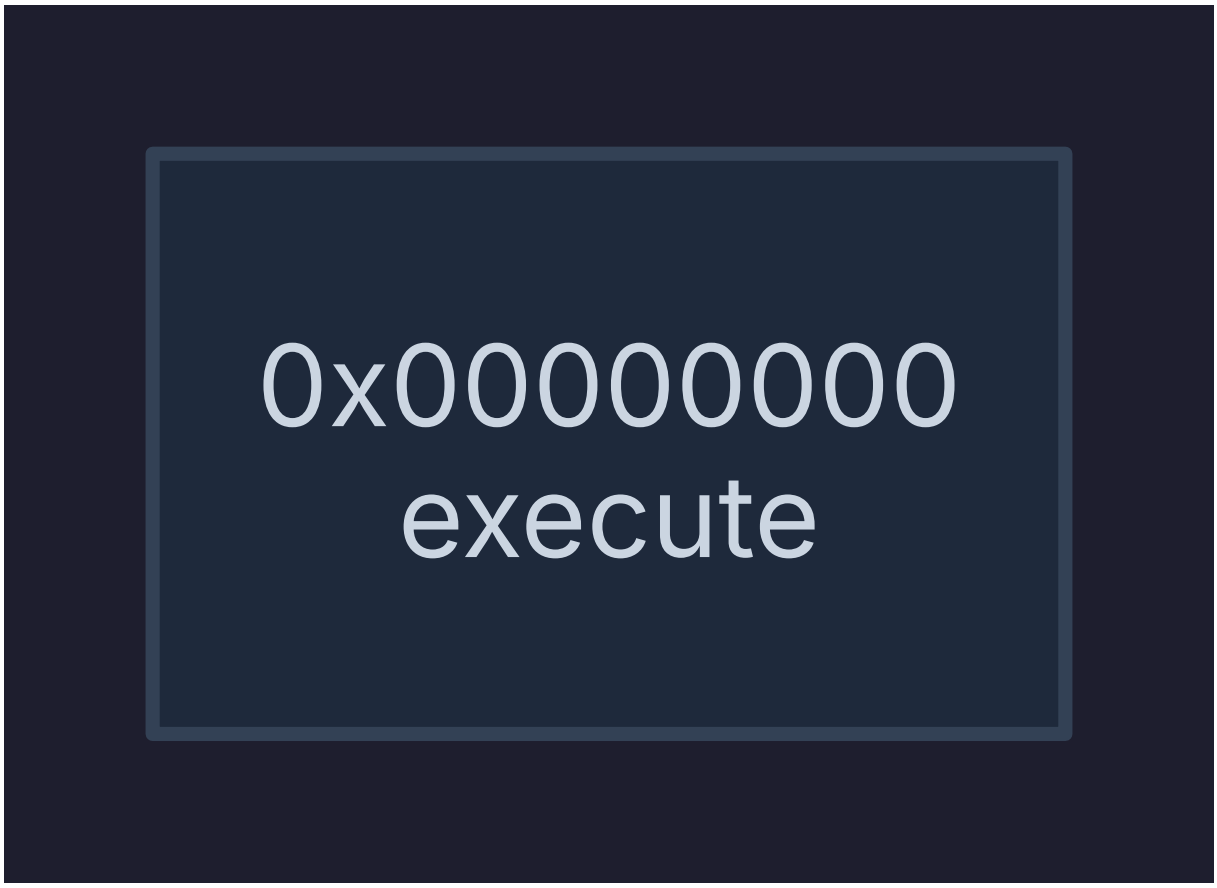
```
// AgentTesla - synthetic analysis
#include <windows.h>
void execute() {
  NtQueryInformationProcess(NULL, NULL, NULL, NULL);
  LoadLibrary(NULL, NULL, NULL, NULL);
  IsDebuggerPresent(NULL, NULL, NULL, NULL);
  ZwSetInformationThread(NULL, NULL, NULL, NULL);
  WSAStartup(NULL, NULL, NULL, NULL);
}
```

Function Call Graph

Edges coloured by API category: file (blue), memory (cyan), process (orange), network (green), registry (yellow), anti-analysis (red), crypto (purple). 2 edges total.



Control Flow Graph (Entry Region)



Loop back-edges shown dashed amber.

Indicators of Compromise

TYPE	INDICATOR	CONF	NOTES
SHA-256	`2e5ff37c3fbba7decb57eea3c94e70e2d383e8fe0fb54aee3ae16a4aea3f73e7`	**HIGH**	primary sample identifier

MITRE ATT&CK Mapping

TACTIC	ID	TECHNIQUE	CONF	OBS
credential-access	**T1555.003**	Credentials from Web Browsers	HIGH	1
defense-evasion	**T1622**	Debugger Evasion	HIGH	1
discovery	**T1057**	Process Discovery	HIGH	1
defense-evasion	**T1055.001**	Dynamic-link Library Injection	HIGH	1
collection	**T1056.001**	Keylogging	HIGH	1

D3FEND Counter-Techniques

D3FEND ID	TACTIC	COUNTER-TECHNIQUE
D3-LAM	Detect	Local Account Monitoring
D3-EAL	Deceive	Emulator/Analyzer Locking
D3-PSA	Detect	Process Spawn Analysis
D3-PSEP	Detect	Process Self-Modification Detection
D3-IBCA	Detect	Input Behaviour Collection Analysis

Detection Rules

YARA**MCA_AgentTesla_2e5ff37c**

Auto-generated from observed import set

```
rule MCA_AgentTesla_2e5ff37c {
  meta:
    author       = "MCA Pipeline"
    description  = "Static signature derived from AgentTesla_2e5ff37c3fbb.bin"
    sha256       = "2e5ff37c3fbba7decb57eea3c94e70e2d383e8fe0fb54aee3ae16a4aea3f73e7"
    family       = "AgentTesla"
    severity     = "auto-derived"
  strings:
    $api00 = "NtQueryInformationProcess" ascii wide
  condition:
    uint16(0) == 0x5A4D and 3 of them
}
```

Recommendations

R1 Block SHA-256 2e5ff37c3fbba7decb57eea3c94e70e2d383e8fe0fb54aee3ae16a4aea3f73e7 at endpoint level.

R2 Hunt across the fleet for the 5 MITRE techniques surfaced in this report — see the ATT&CK Mapping table.

R3 Force credential reset + revoke browser session tokens for any host where this hash is observed.

References

[1] MITRE ATT&CK https://attack.mitre.org
[2] MITRE D3FEND https://d3fend.mitre.org
[3] CISA KEV https://www.cisa.gov/known-exploited-vulnerabilities-catalog
[4] FIRST Malware Analysis Framework https://www.first.org/global/sigs/malware/ma-framework/

Glossary

TERM	DEFINITION
T-number	MITRE ATT&CK technique identifier (T1234 / T1234.001)
TLP	Traffic Light Protocol — AMBER restricts sharing to need-to-know
Family	Malware-family classification, here pre-tagged in the pe_static_v1 corpus