Immunity from antimalware automation attacks

Dennis Batchelder
Hong Jia
October 2013
Call to action

A new bad guy is weaponizing our antimalware products

We’re getting thousands of incoming “crafted” files and suspect telemetry every month

- Probing our automation strategies and signature weak points
- Poisoning our data sources
- Exploiting how we share samples between ourselves

Our industry inadvertently assists the attackers

Let’s work together to fix things before we have a catastrophe
AGENDA

How we got here
The new attacks
The aftermath
Recommendations
How we got here
We automate for good reasons

High malware volume

Short malware lifecycle
Antimalware automation

**Collection**
- Industry and customers
- Automatic and on demand

**Big Data**
- Samples
- Map reduce
- Processed/Workflow

**Analysis**
- Dynamic and Static
- Vendor rescans/determinations
- Human-supplied patterns

**Auto-classification**
- Combine analysis with reputation
- Assign determination, family
- Feeds sig-gen and cloud protection

**Signature Generation**
- Best-fit signature
- Static and proactive
- Signature release pipeline

**Telemetry Monitoring**
- FP detection
- Never unknowns
- Sample requests

**Big Data**
samples, telemetry, reputation, determinations

**Analysis**
Auto-classification

**Signature generation**

**Telemetry response**

Industry
- Samples
- Meta-data
- Reputation
- Determinations

Customers
- Telemetry
- Samples

Map reduce
Processed/Workflow
We know how to handle risks of infrastructure attacks...
<table>
<thead>
<tr>
<th>Risk</th>
<th>Mitigation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Denial of Service blocking samples and telemetry</td>
<td>Collection network protection</td>
</tr>
<tr>
<td>Overload causing slow time to protect</td>
<td>Scale-out architectures</td>
</tr>
<tr>
<td>Analysis exploits taking down/infecting backend systems</td>
<td>Sandboxing, quotas</td>
</tr>
<tr>
<td>Staleness reducing effectiveness</td>
<td>Recency weighting, Curated samples</td>
</tr>
<tr>
<td>Outage</td>
<td>Georedundancy</td>
</tr>
<tr>
<td>FPs</td>
<td>Signature validation pipeline, large clean lists, live monitoring</td>
</tr>
<tr>
<td>Malware infections</td>
<td>Isolation, monitoring</td>
</tr>
<tr>
<td>Malware leakage</td>
<td>Sharing agreements, air gaps, physical security</td>
</tr>
<tr>
<td>PII disclosure</td>
<td>Data cleansing and auditing</td>
</tr>
</tbody>
</table>
But what if data itself is the attack vector?

-what if the sample isn’t sourced from the wild?
-what if incoming telemetry is lying?
-what if the sample is crafted to exploit us?
Risks of data vector attacks

Collection
- Samples
- Meta-data
- Reputation
- Determinations

Big Data
- samples, telemetry, reputation, determinations

Analysis
- Auto-classification

Signature generation

Telemetry response

Attack | Risk
--- | ---
Fake, probe samples | Signature bloat, inefficiency
Automation strategy leakage | Signature weakness leakage
Determination trust leakage | Determination trust leakage
Fake telemetry | Poisoning file reputation
Crafted samples | Signing trigger leakage
| Wide-spread or targeted FPs
| Financial and brand damages
Evil recipe for weaponizing AV products

Learn system weaknesses
- What causes us to accept samples
- How samples spread around the industry
- Which vendor determinations we trust
- What triggers us to use different kinds of signatures
- Holes in our signatures
- Holes in our automation

Launch the attack
- Craft a sample that:
  - Encourages target vendor to sign it
  - Exploits target vendors signature weakness
- Inject sample and telemetry into the system
- Wait, then watch the mess
And why should we care?

Nobody should be able to exploit our systems…

- It hurts our customers
- And damages our reputation

…no matter the motive

- No having fun at our expense
- No embarrassing the security industry
- No preventing us from working together
- No attacks without our knowledge
So, has it happened?
We’ve seen...

Attack sophistication

• Crafted files moving from clean to junk to malicious files
• Use of TOR for sample and telemetry submission

Microsoft-specific targeting

• Discovered an automation strategy weakness and a weak signature type

Broad industry targeting

• Crafted files targeting other AV vendors
• Embedding our (and other) signature fragments as triggers
• Exposing weaknesses in how we exchange samples between ourselves/testers
6 March – 12 April

Assumed goal: automation holes

Method to craft

• Insert signature fragments into clean files’ resource sections
• Submit to VirusTotal via TOR

Results

• ~300 crafted clean files (never seen in wild)
• Many vendors re-sharing and signing
• Our automation treated it as obfuscated sample
• FP with proactive signature on clean code
• Partner FP on copied signature

Learn system weaknesses

- What causes us to accept samples
- How samples spread around the industry
- Which vendor determinations we trust
- What triggers us to use different kinds of signatures
- Holes in our automation
- Holes in our signatures

Launch the attack

- Craft a sample that:
  - Encourages target vendor to sign it
  - Exploits target vendors signature weakness
- Inject sample and telemetry into the system
- Wait, then watch the mess
Crafted clean files

NULLs in .rsr

f7e23305f49a83f5b7ef749c2d8c159b3f7057f9
(Epson Brother file)

Signature Fragment in .rsr

CBDD3071CEB251D84E8B35743A61027C25DE6F66
29 April – present

Assumed goal: signature holes

Method to craft

- Build junk files attempting to cause signature hash collisions
- Insert sig fragment strings/heads to cause “trusted” vendor detections
- Submit to VirusTotal via TOR

Results

- ~2000 crafted junk files (never seen in wild)
- Many vendors re-sharing and signing
- Some vendors sharing with external testers

Learn system weaknesses

- What causes us to accept samples
- How samples spread around the industry
- Which vendor determinations we trust
- What triggers us to use different kinds of signatures
- Holes in our automation
- Holes in our signatures

Launch the attack

- Craft a sample that:
  - Encourages target vendor to sign it
  - Exploits target vendors signature weakness
- Inject sample and telemetry into the system
- Wait, then watch the mess

Assumed goal: signature holes

Method to craft

- Build junk files attempting to cause signature hash collisions
- Insert sig fragment strings/heads to cause “trusted” vendor detections
- Submit to VirusTotal via TOR

Results

- ~2000 crafted junk files (never seen in wild)
- Many vendors re-sharing and signing
- Some vendors sharing with external testers

Crafted junk file

Junk import table

Embedded signature fragments

0x361d9b1375bf5f49f4b9f2f9fc4398d5ffdb353
Crafted junk file, signature collision with malware

“Static” signature collides with Trojan:Win32/Simda

F8A12B809909112BA9E4F175F4D262EE9DEC8DB1

Junk file

a622b580ac5748e0cca17879a303178b118862c0
"Static" signature collides with VisualBoyAdvance

e0a010951cab6bf9bf0d124d7a944e0457cb170
Future (weaponized)

Assumed goal: targeted FP

Method to craft

- Modify real malicious file to cause signature hash collisions with victim clean file
- Compel target vendor to sign with signature fragments from trusted vendor
- Submit to VirusTotal via TOR

Results

- Target vendor signs automatically
- Victim suffers FP against clean file
Our recent investigations

Did we get used as a weapon?

• We searched for an event in past 3 months
  • Static signature weaknesses: searched for inadvertent “test” FPs
  • Nothing conclusive (6 suspicious events)

Is some of our telemetry also crafted?

• We are monitoring TOR-based telemetry
  • 1 out of 100,000 of our endpoints use TOR
  • TOR endpoints seem 4 times as infected as normal users
  • TOR endpoints send one tenth the rate of junk telemetry
  • Nothing found
The aftermath
Changes we’ve made

Big Data
- samples, telemetry, reputation, determinations

Collection
- Industry
  - Samples
  - Meta-data
  - Reputation
  - Determinations
- Customers
  - Telemetry
  - Samples

Analysis
- Auto-classification
- Signature generation
- Telemetry response

Issue
- Signature generation using clean sections when signing crafted clean files
- Static signatures used in automation had CRC collision weakness
- Potential poisoned telemetry
- Not handling artificial escalations very well

Changes
- 1) Auto-detect crafted clean files
- 2) Sign only with static signatures
- 1) Harden signature type to require SHA1 match
- 1) Anomaly monitoring
- 1) Sample sharing requirements to include attestation of sourcing
- 2) Automation rules stop “credit” for detections
- 3) Issue awareness
- 4) Cross-vendor working group
Contaminating AV-Test

2 crafted files showed up in AV-Test’s August testing set

- 0xf019bceae867415dc2027b12b282486973759fa5
- 0x186f720f76bcd6fcc83055a64989ed45cd7b5d66

Andreas Marx investigated

- Vendors give to aggregators
- Aggregators share with testers and vendors
- Testers curate samples, but in the end, they trust vendor sources

Highlights need for vendor control of what is shared

- Artificially inflates the value of these files
- Encourages useless vendor detections
- Could lead to becoming a victim of weaponization
Industry Recommendations

Exchanging unseen samples
  • Causes artificial escalations and drives useless detections
  • *If your customers don’t see it, don’t exchange it*

Automated blind reliance on partner detections
  • detections ≠ determinations
  • *Rely only on vendor samples for vendor determinations*
More Industry Recommendations

Treat this as a serious threat
- Before somebody weaponizes you
- *Find and fix your automation and signature weaknesses*

We need to work together
- *Let’s share crafted file/telemetry awareness and detection/mitigation techniques*