Propagating Threat Scores With a TLS Ecosystem Graph Model Derived by Active Measurements

Markus Sosnowski, Patrick Sattler, Johannes Zirngibl, Tim Betzer, and Georg Carle

Thursday 23rd May, 2024

Network Traffic Measurement and Analysis Conference 2024

Chair of Network Architectures and Services
School of Computation, Information, and Technology
Technical University of Munich
Active Internet-wide DNS and TLS measurements can provide new information on known threats:

- 192.0.2.1
- 203.0.113.1
- evil.corp.org resolves
- evil2.corp.org resolves
- Crt. 1 returns
- corp.org
- parent-sub-domain
- 188.114.96.3

Should we block 203.0.113.1? What about the domains? What about 188.113.96.3?
Active Internet-wide DNS and TLS measurements can provide new information on known threats:

192.0.2.1
203.0.113.1
evil.corp.org
resolves
resolves
Crt. 1
returns
returns
evil2.corp.org
corp.org
parent-
sub-domain
188.114.96.3
IP Address
on a Blocklist
IP Address
Domain
Certificate

Should we block 203.0.113.1?

M. Sosnowski et al. — Propagating Threat Scores With a TLS Ecosystem Graph Model Derived by Active Measurements
Motivation

Active Internet-wide DNS and TLS measurements can provide new information on known threats:

```
192.0.2.1  203.0.113.1
evil.corp.org
resolves
resolves
Crt. 1
returns
returns
evil2.corp.org
corp.org
parent-
sub-domain
188.114.96.3
IP Address
on a Blocklist
IP Address
Domain
Should we block 203.0.113.1?
```
Motivation

Active Internet-wide DNS and TLS measurements can provide new information on known threats:

192.0.2.1 resolves evil.corp.org returns evil2.corp.org
203.0.113.1 resolves corp.org parent-sub-domain 188.114.96.3

IP Address on a Blocklist
IP Address
Domain
Certificate

Should we block 203.0.113.1?

M. Sosnowski et al. — Propagating Threat Scores With a TLS Ecosystem Graph Model Derived by Active Measurements
Motivation

Active Internet-wide DNS and TLS measurements can provide new information on known threats:

IP Address Domain Certificate
192.0.2.1 evil.corp.org Crt. 1
203.0.113.1 evil2.corp.org

Should we block 203.0.113.1? What about the domains?
Motivation

Active Internet-wide DNS and TLS measurements can provide new information on known threats:

Should we block 203.0.113.1? What about the domains? What about 188.113.96.3?
Active Internet-wide DNS and TLS measurements can provide new information on known threats:

Motivation

Should we block 203.0.113.1? What about the domains? What about 188.113.96.3?
Motivation

Challenge: “Internet-wide” is quite large
Motivation

**Challenge:** “Internet-wide” is quite large

An Internet-wide TLS scan from Jan. 2024

<table>
<thead>
<tr>
<th>Type</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domains</td>
<td>628 M</td>
</tr>
<tr>
<td>IPv4 TLS Handshakes</td>
<td>608 M</td>
</tr>
<tr>
<td>IPv6 TLS Handshakes</td>
<td>146 M</td>
</tr>
</tbody>
</table>
Challenge: “Internet-wide” is quite large

An Internet-wide TLS scan from Jan. 2024

<table>
<thead>
<tr>
<th>Type</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domains</td>
<td>628 M</td>
</tr>
<tr>
<td>IPv4 TLS Handshakes</td>
<td>608 M</td>
</tr>
<tr>
<td>IPv6 TLS Handshakes</td>
<td>146 M</td>
</tr>
</tbody>
</table>

⇒

- Any algorithm used on such large datasets has to scale!
- $O(n)$ or faster
Methodology

Modeling the TLS Ecosystem as Graph

Propagating Threat Scores

An Internet-wide TLS Scanning Pipeline
Methodology

Modeling the TLS Ecosystem as Graph

Propagating Threat Scores

An Internet-wide TLS Scanning Pipeline
Modeling the TLS Ecosystem as Graph

- The Internet is a network, modeling collected data as a graph is intuitive
Modeling the TLS Ecosystem as Graph

- The Internet is a network, modeling collected data as a graph is intuitive
- The generalized structure allows applying standard graph algorithms
Modeling the TLS Ecosystem as Graph

- The Internet is a network, modeling collected data as a graph is intuitive
- The generalized structure allows applying standard graph algorithms
- Labeled property graph:
Modeling the TLS Ecosystem as Graph

- The Internet is a network, modeling collected data as a graph is intuitive
- The generalized structure allows applying standard graph algorithms
- Labeled property graph:
  - Data is represented as nodes and edges
Modeling the TLS Ecosystem as Graph

• The Internet is a network, modeling collected data as a graph is intuitive
• The generalized structure allows applying standard graph algorithms
• Labeled property graph:
  • Data is represented as nodes and edges
  • Nodes and edges are labeled and can have arbitrary properties
Modeling the TLS Ecosystem as Graph

- The Internet is a network, modeling collected data as a graph is intuitive
- The generalized structure allows applying standard graph algorithms
- Labeled property graph:
  - Data is represented as nodes and edges
  - Nodes and edges are labeled and can have arbitrary properties
  - Edges are directed
Designing the graph schema:

- Directions in the graph should reflect deliberate actions of the actor controlling a node.
Modeling the TLS Ecosystem as Graph

Designing the graph schema:

• directions in the graph should reflect deliberate actions of the actor controlling a node
Modeling the TLS Ecosystem as Graph

Designing the graph schema:

- directions in the graph should reflect deliberate actions of the actor controlling a node
Modeling the TLS Ecosystem as Graph

Designing the graph schema:

- directions in the graph should reflect deliberate actions of the actor controlling a node

```
<table>
<thead>
<tr>
<th>IP Address</th>
<th>Domain</th>
<th>Certificate</th>
</tr>
</thead>
<tbody>
<tr>
<td>resolves</td>
<td>contains</td>
<td></td>
</tr>
</tbody>
</table>
```
Modeling the TLS Ecosystem as Graph

Designing the graph schema:

- directions in the graph should reflect deliberate actions of the actor controlling a node
Designing the graph schema:

- directions in the graph should reflect deliberate actions of the actor controlling a node
Methodology

Modeling the TLS Ecosystem as Graph

Propagating Threat Scores

An Internet-wide TLS Scanning Pipeline
The Probabilistic Threat Propagation (PTP) [1] algorithm:

- PTP meets our intuition how scores should propagate (considers locality and edge directions)

Propagating Threat Scores

The Probabilistic Threat Propagation (PTP) [1] algorithm:

- PTP meets our intuition how scores should propagate (considers locality and edge directions)
- it’s fast $O(n)$

The Probabilistic Threat Propagation (PTP) [1] algorithm:

- PTP meets our intuition how scores should propagate (considers locality and edge directions)
- it’s fast $O(n)$
- we can use existing blocklists as input

The Probabilistic Threat Propagation (PTP) [1] algorithm:

- PTP meets our intuition how scores should propagate (considers locality and edge directions)
- it’s fast $O(n)$
- we can use existing blocklists as input
- highly connected nodes (e.g., from CDNs) will automatically get low scores

Message-based approximate PTP:

- the input has a fixed score of one (e.g., nodes on a blocklist)
Message-based approximate PTP:

- the input has a fixed score of one (e.g., nodes on a blocklist)
- each node sends its score to neighbors in reversed graph direction
Message-based approximate PTP:

- the input has a fixed score of one (e.g., nodes on a blocklist)
- each node sends its score to neighbors in reversed graph direction
- a node will get the average of received scores as new score
Message-based approximate PTP:

- the input has a fixed score of one (e.g., nodes on a blocklist)
- each node sends its score to neighbors in reversed graph direction
- a node will get the average of received scores as new score
- repeat until convergence
Propagating Threat Scores

Message-based approximate PTP:

- the input has a fixed score of one (e.g., nodes on a blocklist)
- each node sends its score to neighbors in reversed graph direction
- a node will get the average of received scores as new score
- repeat until convergence
Propagating Threat Scores

Message-based approximate PTP:

- the input has a fixed score of one (e.g., nodes on a blocklist)
- each node sends its score to neighbors in reversed graph direction
- a node will get the average of received scores as new score
- repeat until convergence
- core aspect of PTP is to minimize the error introduced by nodes falsely increasing their own score
Propagating Threat Scores

Message-based approximate PTP:

- the input has a fixed score of one (e.g., nodes on a blocklist)
- each node sends its score to neighbors in reversed graph direction
- a node will get the average of received scores as new score
- repeat until convergence
- core aspect of PTP is to minimize the error introduced by nodes falsely increasing their own score
Methodology

Modeling the TLS Ecosystem as Graph

Propagating Threat Scores

An Internet-wide TLS Scanning Pipeline
An Internet-wide TLS Scanning Pipeline

Internet-wide measurements at GINO¹:

- Special interest group since 2016

¹https://net.in.tum.de/projects/gino/
An Internet-wide TLS Scanning Pipeline

Internet-wide measurements at GINO\(^1\):

- Special interest group since 2016
- Among others: Internet-wide DNS, TLS, HTTPS scans on port 443

\(^1\)https://net.in.tum.de/projects/gino/
An Internet-wide TLS Scanning Pipeline

Internet-wide measurements at GINO¹:

- Special interest group since 2016
- Among others: Internet-wide DNS, TLS, HTTPS scans on port 443

¹https://net.in.tum.de/projects/gino/
An Internet-wide TLS Scanning Pipeline

- **various domain sources**
- **DNS scans**
  - *local resolver + massdns*
- **IPv4 & IPv6 port scans**
  - *ZMap, ZMapv6*

Internet-wide measurements at GINO\(^1\):
- Special interest group since 2016
- Among others: Internet-wide DNS, TLS, HTTPS scans on port 443

\(^1\)https://net.in.tum.de/projects/gino/
An Internet-wide TLS Scanning Pipeline

Internet-wide measurements at GINO\(^1\):

- Special interest group since 2016
- Among others: Internet-wide DNS, TLS, HTTPS scans on port 443

\(^1\)https://net.in.tum.de/projects/gino/
An Internet-wide TLS Scanning Pipeline

various domain sources

DNS scans
local resolver + massdns

IPv4 & IPv6 port scans
ZMap, ZMapv6

TLS scans
TUM goscanner

Internet-wide measurements at GINO\(^1\):

- Special interest group since 2016
- Among others: Internet-wide DNS, TLS, HTTPS scans on port 443
- New: Apache spark app to merge our scans and construct the graph model

\(^1\)https://net.in.tum.de/projects/gino/
Results

We created 13 monthly Internet-wide TLS Ecosystem Graphs throughout the last year\(^2\)
We created 13 monthly Internet-wide TLS Ecosystem Graphs throughout the last year\(^2\)

Overview of the latest graph from Jan. 2024

<table>
<thead>
<tr>
<th>Node Type</th>
<th>Amount</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domains</td>
<td>628 M</td>
<td>70.0%</td>
</tr>
<tr>
<td>Certificates</td>
<td>171 M</td>
<td>19.1%</td>
</tr>
<tr>
<td>IPv4 &amp; IPv6 Addresses</td>
<td>98 M</td>
<td>10.9%</td>
</tr>
</tbody>
</table>

\(^2\) starting Jan. 2023
Results

We created 13 monthly Internet-wide TLS Ecosystem Graphs throughout the last year\(^2\)

Overview of the latest graph from Jan. 2024

<table>
<thead>
<tr>
<th>Node Type</th>
<th>Amount</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domains</td>
<td>628 M</td>
<td>70.0%</td>
</tr>
<tr>
<td>Certificates</td>
<td>171 M</td>
<td>19.1%</td>
</tr>
<tr>
<td>IPv4 &amp; IPv6 Addresses</td>
<td>98 M</td>
<td>10.9%</td>
</tr>
</tbody>
</table>

- 90% of edges targeting IP addresses accumulated on only 2% of the nodes

\(^2\) starting Jan. 2023
Results

We created 13 monthly Internet-wide TLS Ecosystem Graphs throughout the last year\(^2\)

Overview of the latest graph from Jan. 2024

<table>
<thead>
<tr>
<th>Node Type</th>
<th>Amount</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domains</td>
<td>628 M</td>
<td>70.0%</td>
</tr>
<tr>
<td>Certificates</td>
<td>171 M</td>
<td>19.1%</td>
</tr>
<tr>
<td>IPv4 &amp; IPv6 Addresses</td>
<td>98 M</td>
<td>10.9%</td>
</tr>
</tbody>
</table>

- 90% of edges targeting IP addresses accumulated on only 2% of the nodes

⇒ we saw a high centralization of the TLS ecosystem, especially for IP addresses

\(^2\) starting Jan. 2023
Results

For each graph and blocklist, we ran the PTP algorithm

<table>
<thead>
<tr>
<th>Blocklist</th>
<th>Observed</th>
</tr>
</thead>
<tbody>
<tr>
<td>abuse.ch Feodo</td>
<td>34</td>
</tr>
<tr>
<td>Blocklist.de Strongips</td>
<td>161</td>
</tr>
<tr>
<td>abuse.ch SSLBL</td>
<td>19</td>
</tr>
<tr>
<td>Openphish</td>
<td>3 461</td>
</tr>
</tbody>
</table>
Results

For each graph and blocklist, we ran the PTP algorithm

<table>
<thead>
<tr>
<th>Blocklist</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>abuse.ch Feodo</td>
<td>C&amp;C IP addresses</td>
</tr>
<tr>
<td>Blocklist.de Strongips</td>
<td>abusive IP addresses</td>
</tr>
<tr>
<td>abuse.ch SSLBL</td>
<td>C&amp;C certificates</td>
</tr>
<tr>
<td>Openphish</td>
<td>phishing domains</td>
</tr>
</tbody>
</table>
For each graph and blocklist, we ran the PTP algorithm

<table>
<thead>
<tr>
<th>Blocklist</th>
<th>Type</th>
<th>Observed</th>
</tr>
</thead>
<tbody>
<tr>
<td>abuse.ch Feodo</td>
<td>C&amp;C IP addresses</td>
<td>34</td>
</tr>
<tr>
<td>Blocklist.de Strongips</td>
<td>abusive IP addresses</td>
<td>161</td>
</tr>
<tr>
<td>abuse.ch SSLBL</td>
<td>C&amp;C certificates</td>
<td>19</td>
</tr>
<tr>
<td>Openphish</td>
<td>phishing domains</td>
<td>3461</td>
</tr>
</tbody>
</table>
Challenge: The Internet is like a black box and we never know if an entity is actually malicious!
Challenge: The Internet is like a black box and we never know if an entity is actually malicious!

- we only have indicators
**Challenge:** The Internet is like a black box and we never know if an entity is actually malicious!

- we only have indicators
- we can show the value of our approach if the identified domains / IP addresses are largely suspicious
How to evaluate whether we found something suspicious?

1. Manual Inspection
2. Comparison with External Threat Intelligence
3. Analysis Over Time
Results

How to evaluate whether we found something **suspicious**?

1. **Manual Inspection**
2. Comparison with External Threat Intelligence
3. Analysis Over Time
Manual Inspection

We quickly noticed several clusters of outliers due to their uniform score and large size:

1. 155k domains resolving to a single IP address with a blocked certificate
2. 38k unbouncepages subdomains
3. 27k sole IP address returning a blocked certificate
4. 3k (seemingly) random domains redirecting to a known phishing domain
Manual Inspection

We quickly noticed several clusters of outliers due to their uniform score and large size

1. 155k domains resolving to a single IP address with a blocked certificate

   figtbnfjxbqjyl[.]in

   IP addr.

   Crt.

   ...

2. 38k unbouncepages subdomains

   unbouncepages[.]com

   5eac214c0fea4e28a5047cb1f4d8b72f[.]unbouncepages[.]com

   04869a1fb7d64442844eaa2b148cadd0[.]unbouncepages[.]com

3. 27k sole IP address returning a blocked certificate

4. 3k (seemingly) random domains redirecting to a known phishing domain

   407979[.]com

   www11666x[.]com

   11666x[.]com

   ...
Manual Inspection

We quickly noticed several clusters of outliers due to their uniform score and large size

1. 155k domains resolving to a single IP address with a blocked certificate

   figtbnfjxbqjyl[.]in
   ... IP addr.
   Crt.

2. 38k unbouncepages subdomains

   04869a1fb7d64442844eaa2b148cadd[.]unbouncepages[.]com
   5eac214c0fea4e28a5047cb1f4d8b72[.]unbouncepages[.]com
   ... unbouncepages[.]com
Manual Inspection

We quickly noticed several clusters of outliers due to their uniform score and large size

1. 155k domains resolving to a single IP address with a blocked certificate

   figtbnfjxbqjyl[.]in
   ...

   IP addr. → Crt.

2. 38k unbouncepages subdomains

   04869a1fb7d64442844eaa2b148cadd0[.]unbouncepages[.]com
   5eac214c0fea4e28a5047cb1f4d8b72f[.]unbouncepages[.]com
   ...

   unbouncepages[.]com

3. 27k sole IP address returning a blocked certificate

M. Sosnowski et al. — Propagating Threat Scores With a TLS Ecosystem Graph Model Derived by Active Measurements
Manual Inspection

We quickly noticed several clusters of outliers due to their uniform score and large size

1. 155k domains resolving to a single IP address with a blocked certificate
   figtbnfjxbqjyl[.]in
   IP addr.
   Crt.

2. 38k unbouncepages subdomains
   04869a1fb7d64442844eaa2b148cadd[.]unbouncepages[.]com
   5eac214c0fea4e28a5047cb1f4d8b72f[.]unbouncepages[.]com
   unbouncepages[.]com
   ...

3. 27k sole IP address returning a blocked certificate

4. 3k (seemingly) random domains redirecting to a known phishing domain
   11666x[.]com
   www11666x[.]com
   407979[.]com
   ...

M. Sosnowski et al. — Propagating Threat Scores With a TLS Ecosystem Graph Model Derived by Active Measurements
How to evaluate whether we found something **suspicious**?

1. Manual Inspection
2. **Comparison with External Threat Intelligence**
3. Analysis Over Time
Comparison with External Threat Intelligence

Threat intelligence services:

- Provide API to check a domain or IP address
Comparison with External Threat Intelligence

Threat intelligence services:

- Provide API to check a domain or IP address
- VirusTotal (VT)\(^3\)
  - aggregates a large amount of threat intelligence feeds (e.g., blocklists)

\(^3\)https://www.virustotal.com
Comparison with External Threat Intelligence

Threat intelligence services:

- Provide API to check a domain or IP address
- VirusTotal (VT)$^3$
  - aggregates a large amount of threat intelligence feeds (e.g., blocklists)
- Google Safe Browsing (GSB)$^4$
  - threats information detected by Google

---

$^3$https://www.virustotal.com
$^4$https://safebrowsing.google.com
Comparison with External Threat Intelligence

Threat intelligence services:

- Provide API to check a domain or IP address
- VirusTotal (VT)\(^3\)
  - aggregates a large amount of threat intelligence feeds (e.g., blocklists)
- Google Safe Browsing (GSB)\(^4\)
  - threats information detected by Google
- However, both have a very rate-limited API

---

\(^3\)https://www.virustotal.com
\(^4\)https://safebrowsing.google.com
Comparison with External Threat Intelligence

Domains with a PTP score above the threshold\(^5\) (without the first three manually identified clusters):

\[\text{Threshold} \quad \text{Domains [k]}\]

\(\text{VT & GSB Categories}
- malicious
- harmless
- unknown\]

\(^5\)only the latest graph from Jan. 2024
Comparison with External Threat Intelligence

IP Addresses with a PTP score above the threshold\(^6\) (without the first three manually identified clusters):

![Graph showing comparison with external threat intelligence]

\(^6\) only the latest graph from Jan. 2024
Results

How to evaluate whether we found something **suspicious**?

1. Manual Inspection
2. Comparison with External Threat Intelligence
3. *Analysis Over Time*
Analysis Over Time

- **Reminder:** We created monthly graphs over the last year
• **Reminder:** We created monthly graphs over the last year
• For each graph, we can evaluate whether nodes with a high score appeared later on the same blocklist
• **Reminder:** We created monthly graphs over the last year
• For each graph, we can evaluate whether nodes with a high score appeared later on the same blocklist
• Calculating the **Appearance Rate:**

1. run PTP with a single blocklist as input
2. identify nodes with high scores
3. calculate portion of nodes appearing later in time on input blocklist
Nodes with a score above an optimized threshold and the portion appearing later on the same blocklist
Nodes with a score above an *optimized* threshold and the portion appearing later on the same blocklist.
We offer an approach that can navigate the millions of possible domains and IP addresses, to help security researchers focus on suspicious subsets of the Internet when searching for unknown threats.

Read our paper! We provide:

- a versatile TLS ecosystem graph model built around deliberate actions
- a PTP algorithm to propagate threat scores
- three analyses that highlight how our approach focuses on malicious activity
- published results, interactive plots, scripts, and code

https://tumi8.github.io/iteg/
• loading the graph model in Neo4J allows to quickly explore server infrastructure
• did you knew ifip.org is also hosted under ifip.or.at, although TMA only under tma.ifip.org?
• loading the neighbors of ifip.org would reveal many more IFIP conferences
Appendix
Example - Early Detection of a Domain

- our graph loaded into Neo4J for easy manual navigation
- only `usps[].trackmypkg-servi[].shop`, `usps[].logistic-mypkg[].shop`, and `usps[].speed-mypkg[].shop` were blocked by OpenPhish
- `bluewishlists[].shop` appeared later on the blocklist (threat score 67%)
- `usps[].logistic-info[].shop` never appeared on the list
Appendix

Optimizing the Detection Threshold

Best performing thresholds:

- Domains: 51%
- IP addresses: 18%
Centralization of the TLS Ecosystem

The diagram illustrates the cumulative portion of nodes versus accumulated edges across different categories: domains, IP addresses, and certificates.