ALPC is in Danger: ALPChecker Detects Spoofing and Blinding Attacks

Anastasiia Kropova, Igor Korkin, Ph.D.
Who we are

Anastasiia Kropova
- Bachelor of Cyber Security
- alumni of NRNU MEPhI (Moscow Engineering Physics Institute)
- Cryptology and Cybersecurity Department

Igor Korkin, Ph.D.
- Independent Security Researcher
- Speaker at BlackHat, HITB, CDFSL, SADFE
- sites.google.com/site/igorkorkin
Interprocess communication in Windows

ALPC client

Process - 1

ALPC

Process - 2

ALPC server

Interprocess communication cases:
- Usage of dynamic-linked libraries;
- Verification of the authenticity of the user to perform the operation on his behalf;
- Getting input text;
- Creation or removal of threads;
- D-COM-object interaction.
Interprocess communication in Windows – ALPC architecture

ALPC client

Process - 1

ALPC

ALPC server

Process - 2

Client Communication Port

Connection Port

Server Communication Port

ALPC ports

User mode

Kernel mode
### ALPC port structures

<table>
<thead>
<tr>
<th>ALPC_PORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>PortListEntry</td>
</tr>
<tr>
<td>CommunicationInfo</td>
</tr>
<tr>
<td>OwnerProcess</td>
</tr>
<tr>
<td>PendingQueue</td>
</tr>
<tr>
<td>WaitQueue</td>
</tr>
<tr>
<td>CanceledQueue</td>
</tr>
<tr>
<td>MainQueue</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ALPC_COMMUNICATION_INFO</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConnectionPort</td>
</tr>
<tr>
<td>ClientCommunicationPort</td>
</tr>
<tr>
<td>ServerCommunicationPort</td>
</tr>
<tr>
<td>HandleTable</td>
</tr>
<tr>
<td>CloseMessage</td>
</tr>
</tbody>
</table>

### C++ Code Snippet

```cpp
PortListEntry
CommunicationInfo
OwnerProcess
PendingQueue
WaitQueue
CanceledQueue
MainQueue
```

### C Code Snippet

```c
PortListEntry
CommunicationInfo
OwnerProcess
PendingQueue
WaitQueue
CanceledQueue
MainQueue
```

### Assembly Code Snippet

```assembly
PortListEntry
CommunicationInfo
OwnerProcess
PendingQueue
WaitQueue
CanceledQueue
MainQueue
```
Kernel mode attacks using Windows Kernel Drivers

Kernel driver attacks in 2022-2023

- Bring your own vulnerable driver (BYOVD)
- Malware drivers signed with leaked certificates
- UEFI security threats

- Pogonin and Korkin at the ADFSL and Rootcon conferences in 2022 gave more than 50 attack examples, accomplished using these techniques;
- Rapid7 experts gave 30 malware examples that use buggy signed drivers (2022);
- TrendMicro experts analyzed low-level threads evolution in Windows and made a conclusion about actuality of kernel-level threads (2023)

Binarly team guys first payed attention to ALPC vulnerability. In 2022 at LABScon and Ekoparty conferences they demonstrated attack on WMI based on disabling ALPC connection.
Test bench – 2 client apps and 2 server apps interacting via ALPC

- Client - 1
- Server - 1
- Client - 2
- Server - 2

User mode

- Client Communication Port C1
- Server Communication Port S1
- Client Communication Port C2
- Server Communication Port S2

Kernel mode

- Connection Port S1
- Connection Port S2
Attack №1 via client port modification

Client - 1
Client - 2
Server - 1
Server - 2

User mode
Kernel mode

Client Communication Port C1
Client Communication Port C2
Server Communication Port S1
Server Communication Port S2
Connection Port S1
Connection Port S2
Attack №2 via server port modification

Client - 1

Client - 2

Server - 1

Server - 2

User mode

Kernel mode

Client Communication Port C1

Client Communication Port C2

Server Communication Port S1

Connection Port S1

Connection Port S2

Server Communication Port S2
Attack №3 via client port modification and server termination (1/3)

Client - 1 → Client - 2 → Server - 1

User mode

Client Communication Port C1

Client Communication Port C2

Server Communication Port S1

Connection Port S1

Server - 2

Kernel mode

Server Communication Port S2

Connection Port S2
Attack №3 via client port modification and server termination (2/3)

Client - 1 → Server - 1

User mode

Client - 2

Client Communication Port C1 → Connection Port S1

Kernel mode

Client Communication Port C2

Server Communication Port S1 → Server Communication Port S2

Connection Port S1 → Server Communication Port S2

Server - 2

Connection Port S2
Attack №3 via client port modification and server termination (3/3)

Client - 1

Client - 2

Server - 1

Server - 2

User mode

Kernel mode

Client Communication Port C1

Client Communication Port C2

Server Communication Port S1

Connection Port S1

Connection Port S2

Server Communication Port S2
# Attacks results

<table>
<thead>
<tr>
<th>Attack</th>
<th>Processes with modified structures</th>
<th>Modified fields</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>via client port modification</td>
<td>Client C2</td>
<td>ClientCommunicationPort -&gt;CommunicationInfo-&gt; ConnectionPort</td>
<td>Illegal connection C2-S1 has been established, S2 lost connection with C2</td>
</tr>
<tr>
<td>via server port modification</td>
<td>Server S2</td>
<td>ServerCommunicationPort -&gt;CommunicationInfo-&gt; ConnectionPort</td>
<td>Illegal connection C2-S1 has been established, S2 lost connection with C2</td>
</tr>
<tr>
<td>via client port modification and server termination</td>
<td>Client C2</td>
<td>ClientCommunicationPort -&gt; CommunicationInfo-&gt; ConnectionPort, ClientCommunicationPort -&gt; CommunicationInfo-&gt; ServerCommunicationPort</td>
<td>Illegal connection C2-S1 has been established, after terminating S2 connection was restored</td>
</tr>
</tbody>
</table>
ALPChecker - a tool for detecting attacks on ALPC interaction

![ALPChecker Diagram](image)
ALPChecker algorithm

begin

Collect information about ALPC connections in the system and distribute them by client and server connections lists

For each client connection from one of the lists

Connection with appropriate ports in other list found

Yes

Add the connection to the Suspicious connections list

No

Show the security message about possible attacks

end
**ALPChecker results: all the attacks were detected**

<table>
<thead>
<tr>
<th>Attack</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>via client port modification</td>
<td>Detected</td>
</tr>
<tr>
<td>via server port modification</td>
<td>Detected</td>
</tr>
<tr>
<td>via client port modification and server termination</td>
<td>Detected</td>
</tr>
</tbody>
</table>
THANK YOU!

Anastasiia Kropova
kropovaanastasiia@gmail.com
github.com/AnastasiKro/ALPChecker

Igor Korkin
igor.korkin@gmail.com
sites.google.com/site/igorkorkin