Exploring Ancient Ruins to Find Modern Bugs: Discovering a 0-Day in MS-RPC Service

Ben Barnea & Ophir Harpaz

whoweare

Ben Barnea
Security Researcher
Akamai

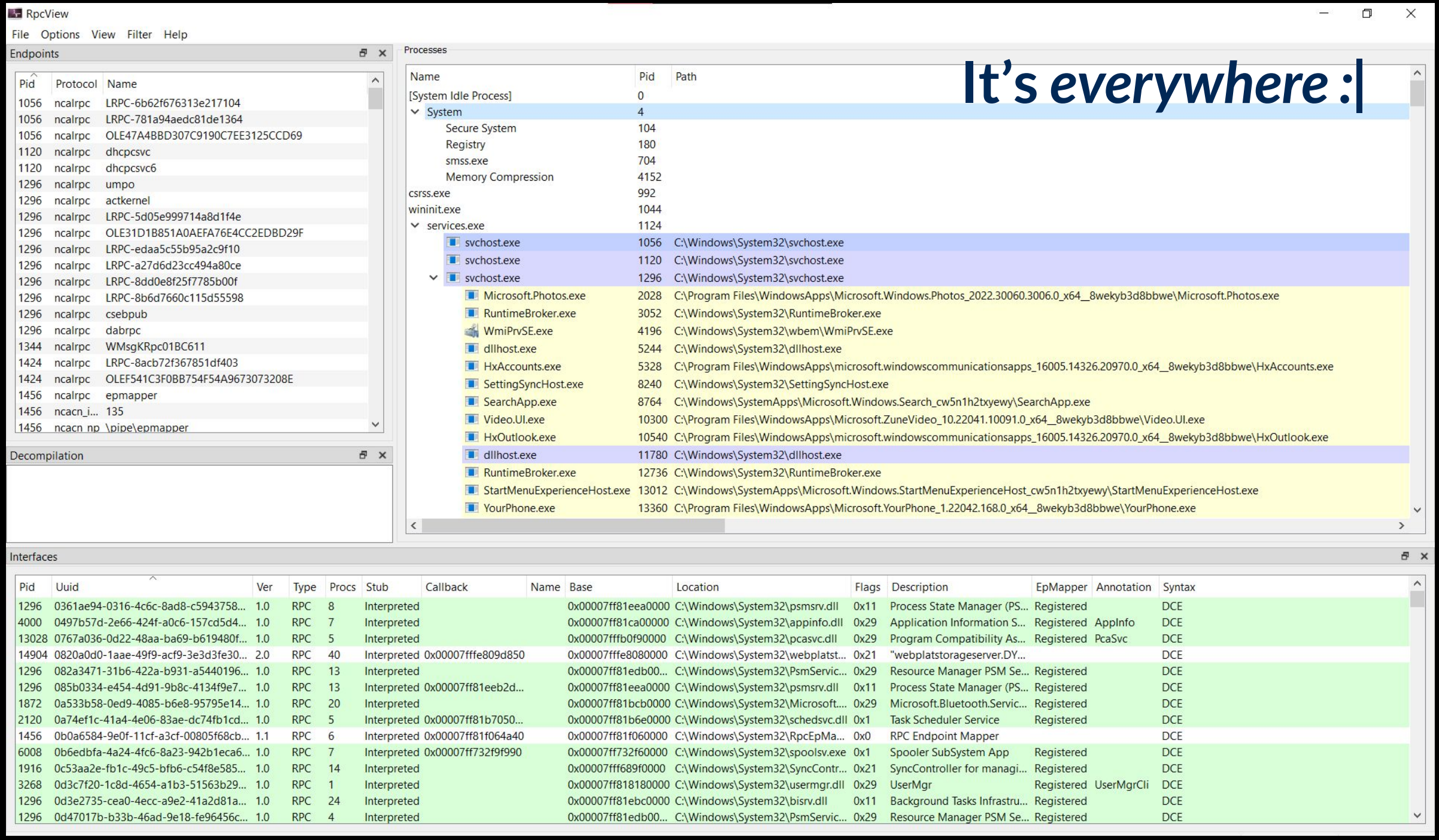
@nachoskrnl

Ophir Harpaz
Security Research team lead
Akamai

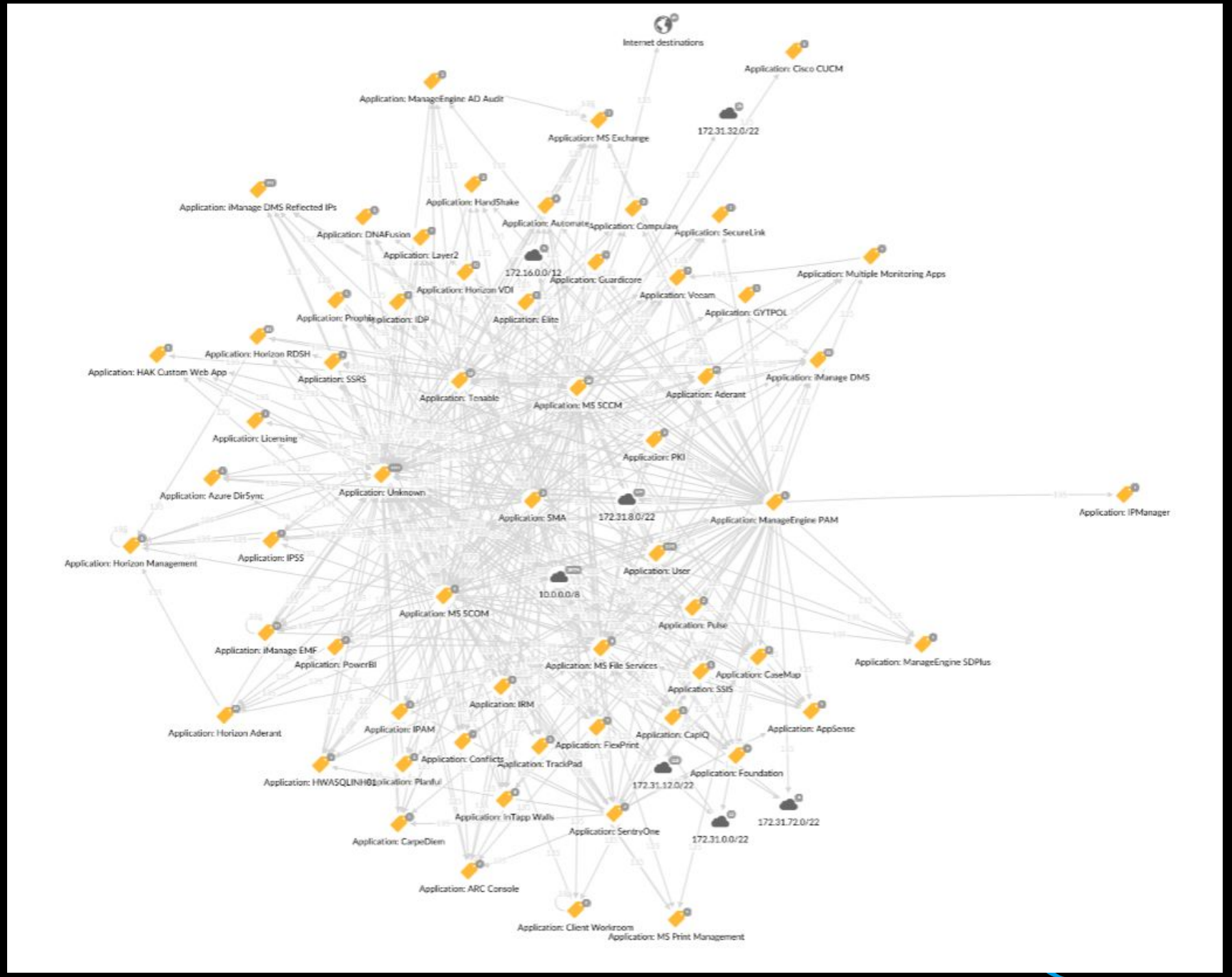
@OphirHarpaz

Akamai

Why MS-RPC?


RpcView
File Options View Filter Help
Endpoints
Pid Protocol Name
1056 ncalrpc LRPC-6b62f676313e217104
1056 ncalrpc LRPC-781a94aedc81de1364
1056 ncalrpc OLE47A4BBD307C9190C7EE3125CCD69
1120 ncalrpc dhcpcsvc
1120 ncalrpc dhcpcsvc6
1296 ncalrpc umpo
1296 ncalrpc actkernel
1296 ncalrpc LRPC-5d05e999714a8d1f4e
1296 ncalrpc OLE31D1B851A0AEFA76E4CC2EDBD29F
1296 ncalrpc LRPC-edaa5c55b95a2c9f10
1296 ncalrpc LRPC-a27d6d23cc494a80ce
1296 ncalrpc LRPC-8dd0e8f25f7785b00f
1296 ncalrpc LRPC-8b6d7660c115d55598
1296 ncalrpc csebpub
1296 ncalrpc dabrpc
1344 ncalrpc WMsgKRpc01BC611
1424 ncalrpc LRPC-8acb72f367851df403
1424 ncalrpc OLEF541C3F0BB754F54A9673073208E
1456 ncalrpc epmapper
1456 ncacn_i... 135
1456 ncacn_np \pipe\epmapper
Decompilation
Processes
It's everywhere :|
Name Pid Path
[System Idle Process] 0
System 4
Secure System 104
Registry 180
smss.exe 704
Memory Compression 4152
csrss.exe 992
wininit.exe 1044
services.exe 1124
svchost.exe 1056 C:\Windows\System32\svchost.exe
svchost.exe 1120 C:\Windows\System32\svchost.exe
svchost.exe 1296 C:\Windows\System32\svchost.exe
Microsoft.Photos.exe 2028 C:\Program Files\WindowsApps\Microsoft.Windows.Photos_2022.30060.3006.0_x64__8wekyb3d8bbwe\Microsoft.Photos.exe
RuntimeBroker.exe 3052 C:\Windows\System32\RuntimeBroker.exe
WmiPrvSE.exe 4196 C:\Windows\System32\wbem\WmiPrvSE.exe
dllhost.exe 5244 C:\Windows\System32\dllhost.exe
HxAccounts.exe 5328 C:\Program Files\WindowsApps\microsoft.windowscommunicationsapps_16005.14326.20970.0_x64__8wekyb3d8bbwe\HxAccounts.exe
SettingSyncHost.exe 8240 C:\Windows\System32\SettingSyncHost.exe
SearchApp.exe 8764 C:\Windows\SystemApps\Microsoft.Windows.Search_cw5n1h2txyewy\SearchApp.exe
Video.UI.exe 10300 C:\Program Files\WindowsApps\Microsoft.ZuneVideo_10.22041.10091.0_x64__8wekyb3d8bbwe\Video.UI.exe
HxOutlook.exe 10540 C:\Program Files\WindowsApps\microsoft.windowscommunicationsapps_16005.14326.20970.0_x64__8wekyb3d8bbwe\HxOutlook.exe
dllhost.exe 11780 C:\Windows\System32\dllhost.exe
RuntimeBroker.exe 12736 C:\Windows\System32\RuntimeBroker.exe
StartMenuExperienceHost.exe 13012 C:\Windows\SystemApps\Microsoft.Windows.StartMenuExperienceHost_cw5n1h2txyewy\StartMenuExperienceHost.exe
YourPhone.exe 13360 C:\Program Files\WindowsApps\Microsoft.YourPhone_1.22042.168.0_x64__8wekyb3d8bbwe\YourPhone.exe
Interfaces
Pid Uuid Ver Type Procs Stub Callback Name Base Location Flags Description EpMapper Annotation Syntax
1296 0361ae94-0316-4c6c-8ad8-c5943758... 1.0 RPC 8 Interpreted 0x00007ff81eea0000 C:\Windows\System32\psmsrv.dll 0x11 Process State Manager (PS... Registered DCE
4000 0497b57d-2e66-424f-a0c6-157cd5d4... 1.0 RPC 7 Interpreted 0x00007ff81ca00000 C:\Windows\System32\appinfo.dll 0x29 Application Information S... Registered AppInfo DCE
13028 0767a036-0d22-48aa-ba69-b619480f... 1.0 RPC 5 Interpreted 0x00007fffb0f90000 C:\Windows\System32\pcasvc.dll 0x29 Program Compatibility As... Registered PcaSvc DCE
14904 0820a0d0-1aae-49f9-acf9-3e3d3fe30... 2.0 RPC 40 Interpreted 0x00007fffe809d850 0x00007fffe8080000 C:\Windows\System32\webplatst... 0x21 "webplatstorageserver.DY... DCE
1296 082a3471-31b6-422a-b931-a5440196... 1.0 RPC 13 Interpreted 0x00007ff81edb00... C:\Windows\System32\PsmServic... 0x29 Resource Manager PSM Se... Registered DCE
1296 085b0334-e454-4d91-9b8c-4134f9e7... 1.0 RPC 13 Interpreted 0x00007ff81eeb2d... 0x00007ff81eea0000 C:\Windows\System32\psmsrv.dll 0x11 Process State Manager (PS... Registered DCE
1872 0a533b58-0ed9-4085-b6e8-95795e14... 1.0 RPC 20 Interpreted 0x00007ff81bcb0000 C:\Windows\System32\Microsoft.... 0x29 Microsoft.Bluetooth.Servic... Registered DCE
2120 0a74ef1c-41a4-4e06-83ae-dc74fb1cd... 1.0 RPC 5 Interpreted 0x00007ff81b7050... 0x00007ff81b6e0000 C:\Windows\System32\schedsvc.dll 0x1 Task Scheduler Service Registered DCE
1456 0b0a6584-9e0f-11cf-a3cf-00805f68cb... 1.1 RPC 6 Interpreted 0x00007ff81f064a40 0x00007ff81f060000 C:\Windows\System32\RpcEpMa... 0x0 RPC Endpoint Mapper DCE
6008 0b6edbfa-4a24-4fc6-8a23-942b1eca6... 1.0 RPC 7 Interpreted 0x00007ff732f9f990 0x00007ff732f60000 C:\Windows\System32\spoolsv.exe 0x1 Spooler SubSystem App Registered DCE
1916 0c53aa2e-fb1c-49c5-bfb6-c54f8e585... 1.0 RPC 14 Interpreted 0x00007fff689f0000 C:\Windows\System32\SyncContr... 0x21 SyncController for managi... Registered DCE
3268 0d3c7f20-1c8d-4654-a1b3-51563b29... 1.0 RPC 1 Interpreted 0x00007ff818180000 C:\Windows\System32\usermgr.dll 0x29 UserMgr Registered UserMgrCli DCE
1296 0d3e2735-cea0-4ecc-a9e2-41a2d81a... 1.0 RPC 24 Interpreted 0x00007ff81ebc0000 C:\Windows\System32\bisrv.dll 0x11 Background Tasks Infrastru... Registered DCE
1296 0d47017b-b33b-46ad-9e18-fe96456c... 1.0 RPC 4 Interpreted 0x00007ff81edb00... C:\Windows\System32\PsmServic... 0x29 Resource Manager PSM Se... Registered DCE

… and between everyone



Yet not much public research

Most information boils down to:

- MSFT documentation
- Several research-oriented blog posts
- Few public vulnerabilities

Why so?

Potential impact:

Lateral Movement & Privilege Escalation

Our agenda for today

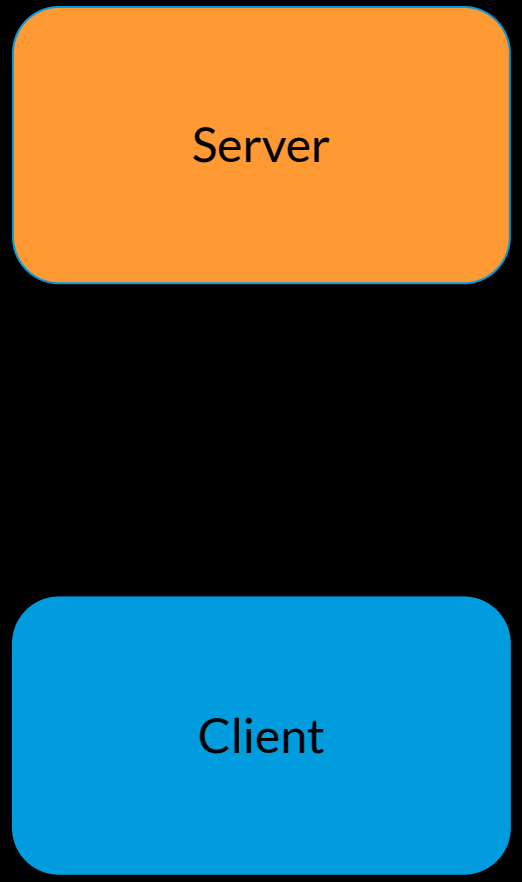
- MS-RPC introduction and overview
- MS-RPC (in)security
- A 0-day in a Windows service

MS-RPC Overview

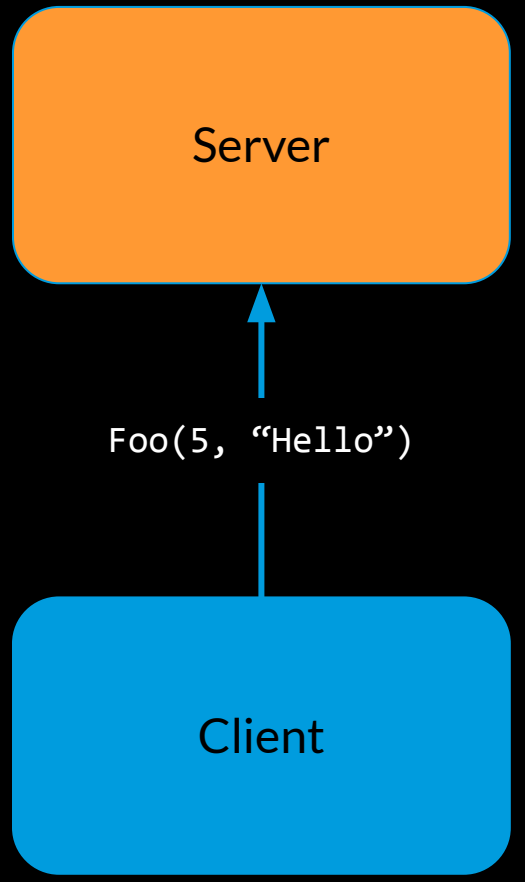
Terminology you'll soon master

- Interface
- {M}IDL
- Transport
- Endpoint
- Binding

The RPC Client-Server Model



The RPC Client-Server Model

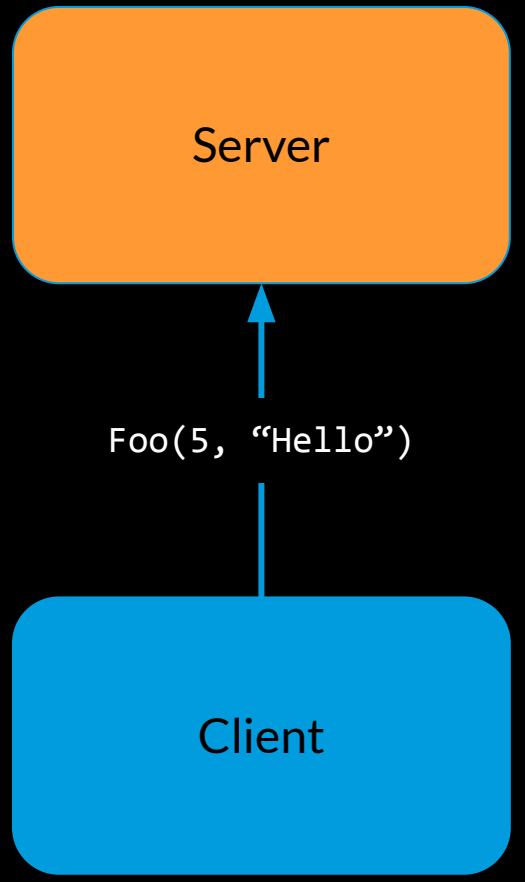


Akamai

The RPC Client-Server Model

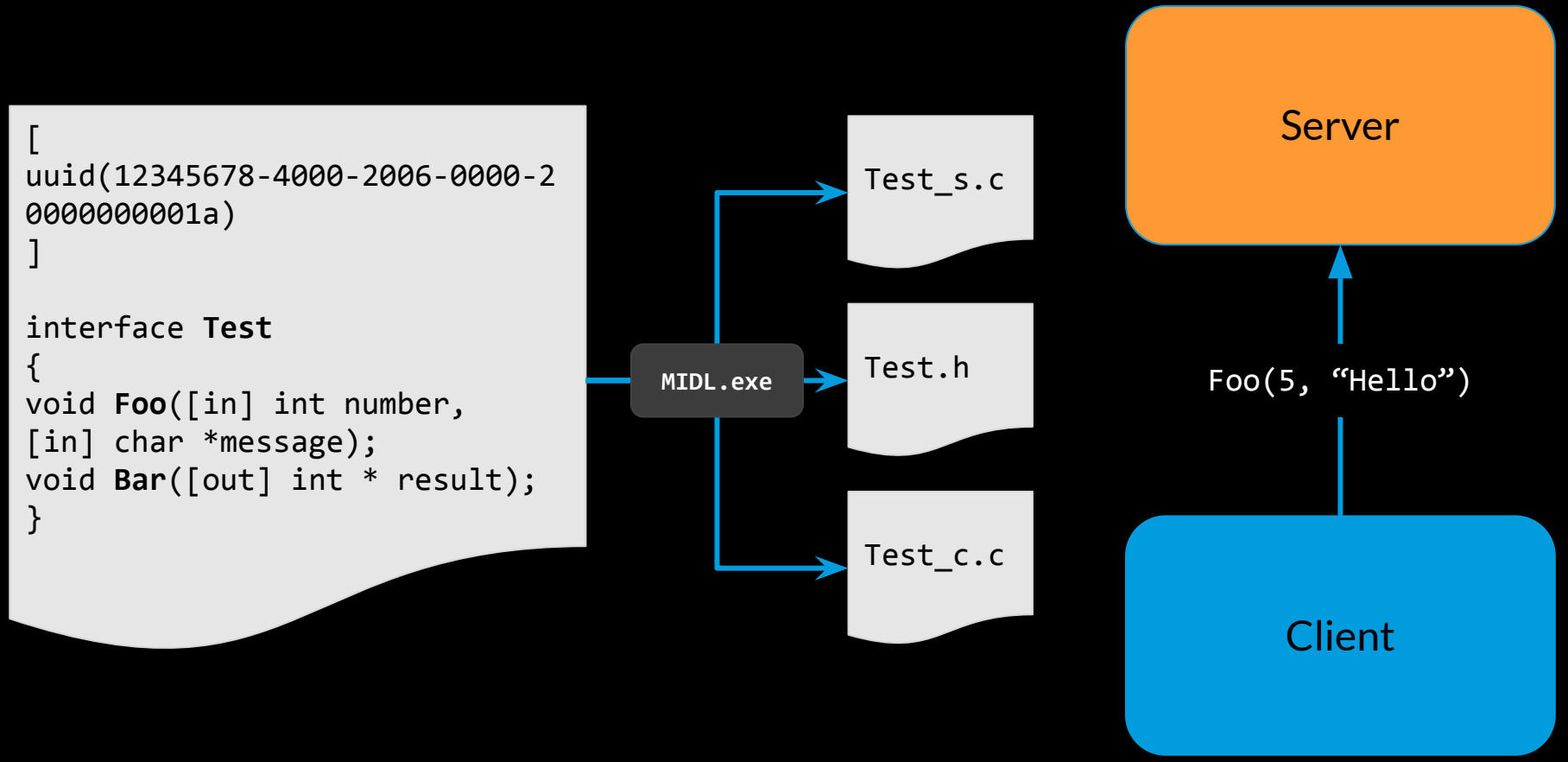
```
[
uuid(12345678-4000-2006-0000-2
0000000001a)
]

interface Test
{
void Foo([in] int number,
[in] char *message);
void Bar([out] int * result);
}
```



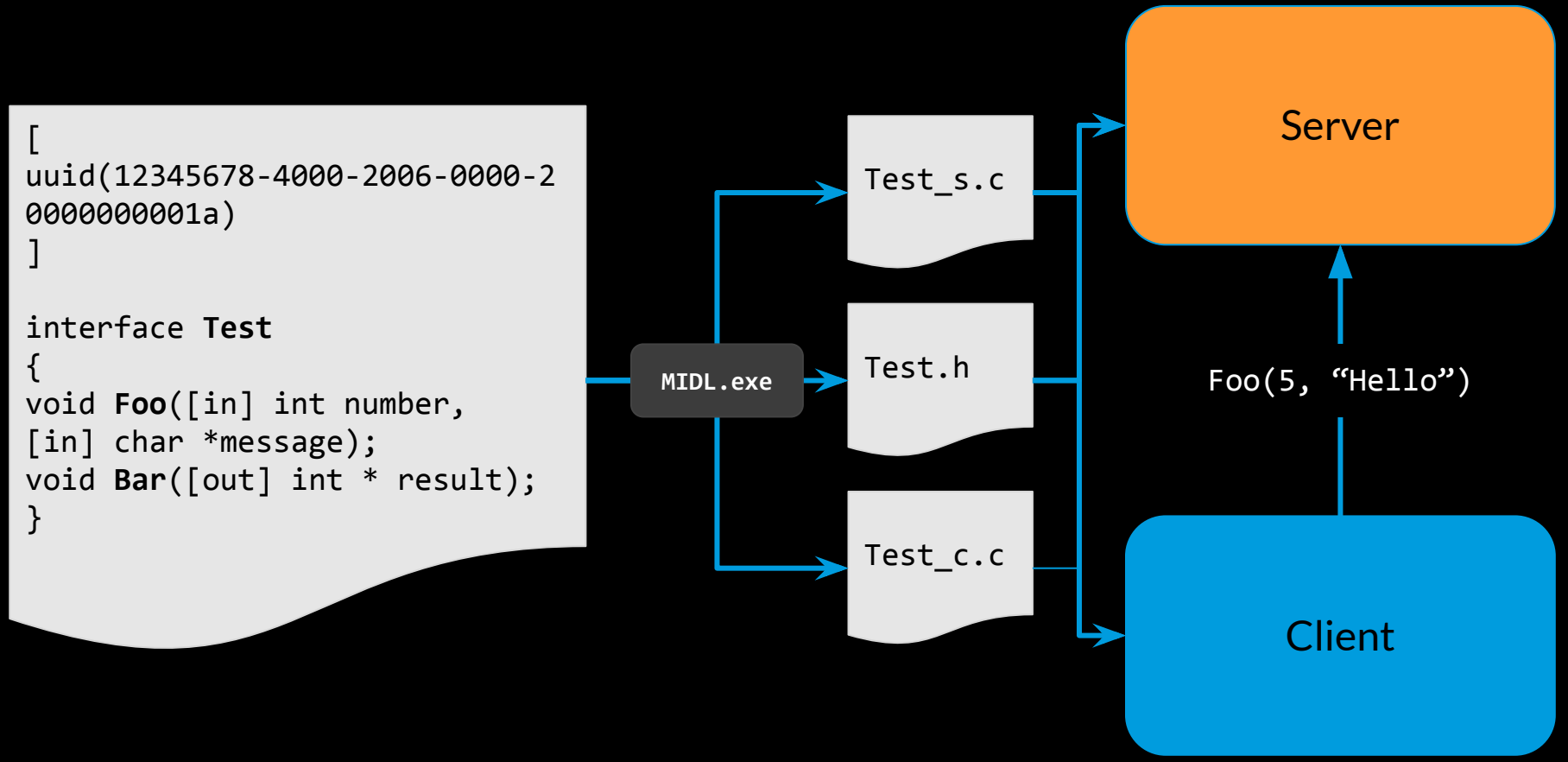
Akamai

The RPC Client-Server Model



Akamai

The RPC Client-Server Model



Akamai

Endpoints

- The server registers an *endpoint* using a certain *transport*

Transports	Protocol Sequence	Endpoints
TCP	`ncacn_ip_tcp`	<port number>
Named pipe	`ncacn_np`	<pipe name>
UDP	`ncadg_ip_udp`	<port number>
ALPC	`ncalrpc`	<ALPC port>
HTTP	`ncacn_http`	<hostname>
Hyper-V socket	`ncacn_hvsocket`	<UUID>

- The endpoint is not bound to an interface

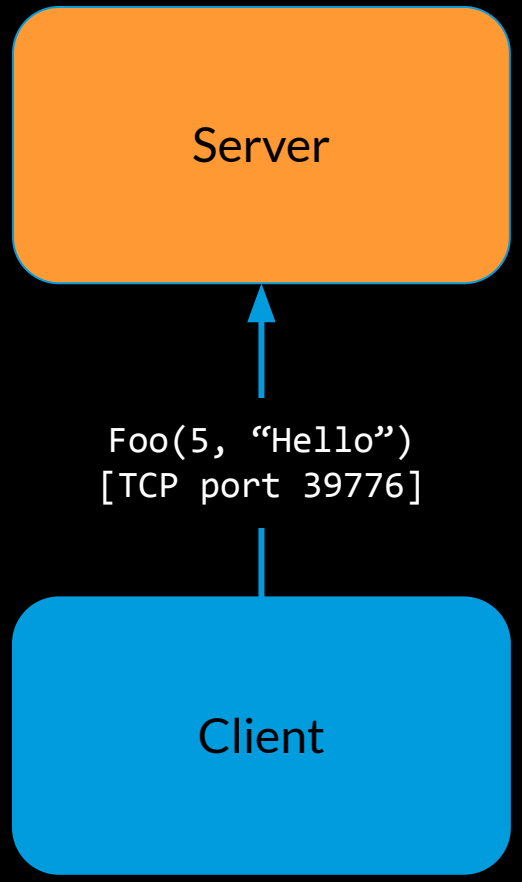
Endpoint Examples

Endpoints

Pid	Protocol	Name
1260	ncacn_hvsocket	DA32E281-383E-49A1-900A-AF3B74B90B0E
1260	ncacn_ip_tcp	135
5516	ncacn_ip_tcp	4290
876	ncacn_ip_tcp	49666
2008	ncacn_ip_tcp	49667
5176	ncacn_ip_tcp	49668
2008	ncacn_np	\PIPE\atsvc
1260	ncacn_np	\pipe\epmapper
876	ncacn_np	\pipe\eventlog
7828	ncacn_np	\PIPE\ROUTER
5176	ncacn_np	\pipe\spoolss
6260	ncacn_np	\PIPE\srvsvc
6472	ncacn_np	\pipe\trkwks
1664	ncacn_np	\PIPE\W32TIME_ALT
5504	ncacn_np	\PIPE\wkssvc
25284	ncalrpc	5c2165c5-bbfa-4a23-85b9-da7cc736639c
1148	ncalrpc	actkernel
6028	ncalrpc	AppV-ISV-APPV-jitv_server
6028	ncalrpc	AppV-ISV-f432e7a9-769f-460c-a3fe-7de4ed58ed3...
6028	ncalrpc	AppV-ISV-f432e7a9-769f-460c-a3fe-7de4ed58ed3...
6028	ncalrpc	AppV-ISV-f432e7a9-769f-460c-a3fe-7de4ed58ed3...
6028	ncalrpc	AppV-ISV-f432e7a9-769f-460c-a3fe-7de4ed58ed3...

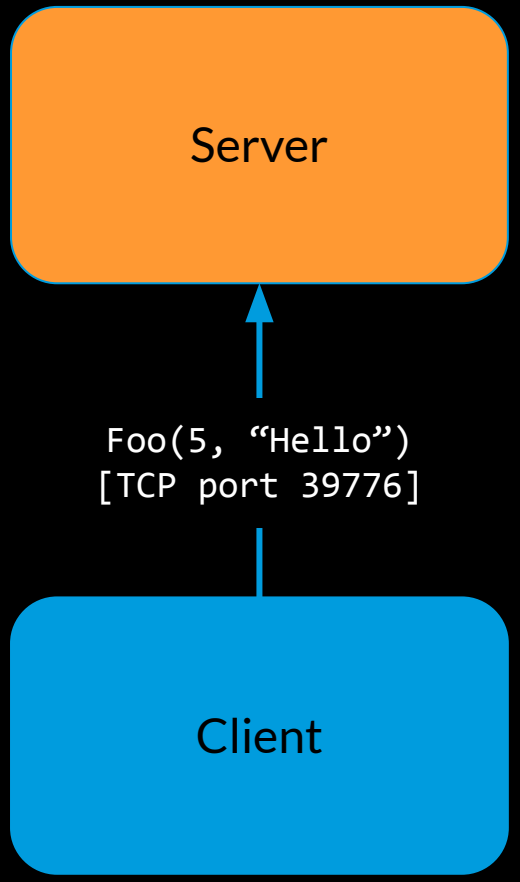
Well-Known Endpoints

Dynamic Endpoints

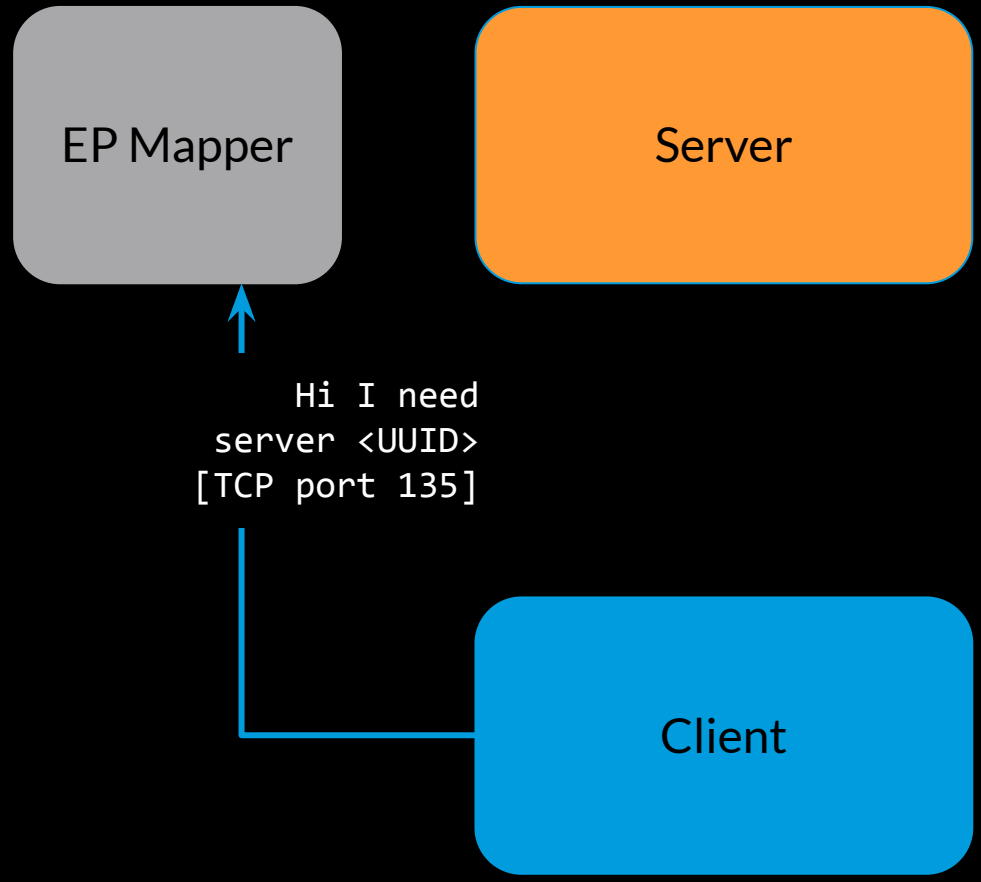


Akamai

Well-Known Endpoints

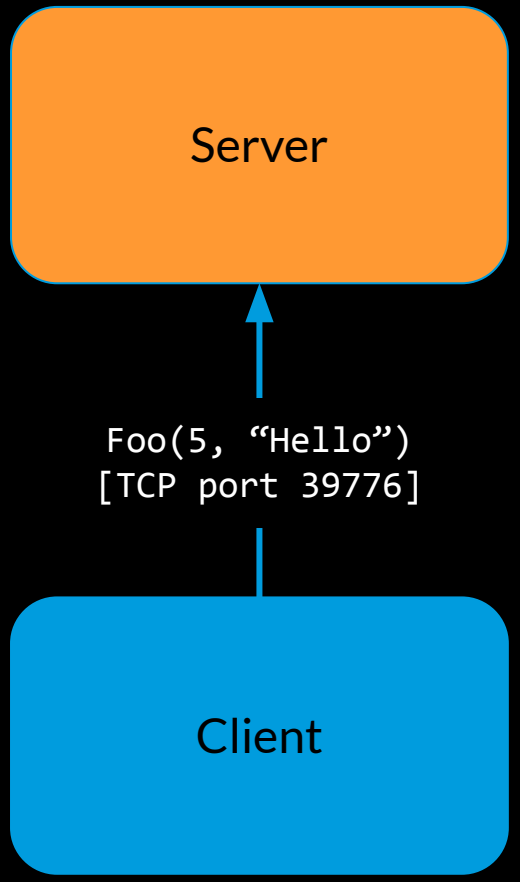


Dynamic Endpoints

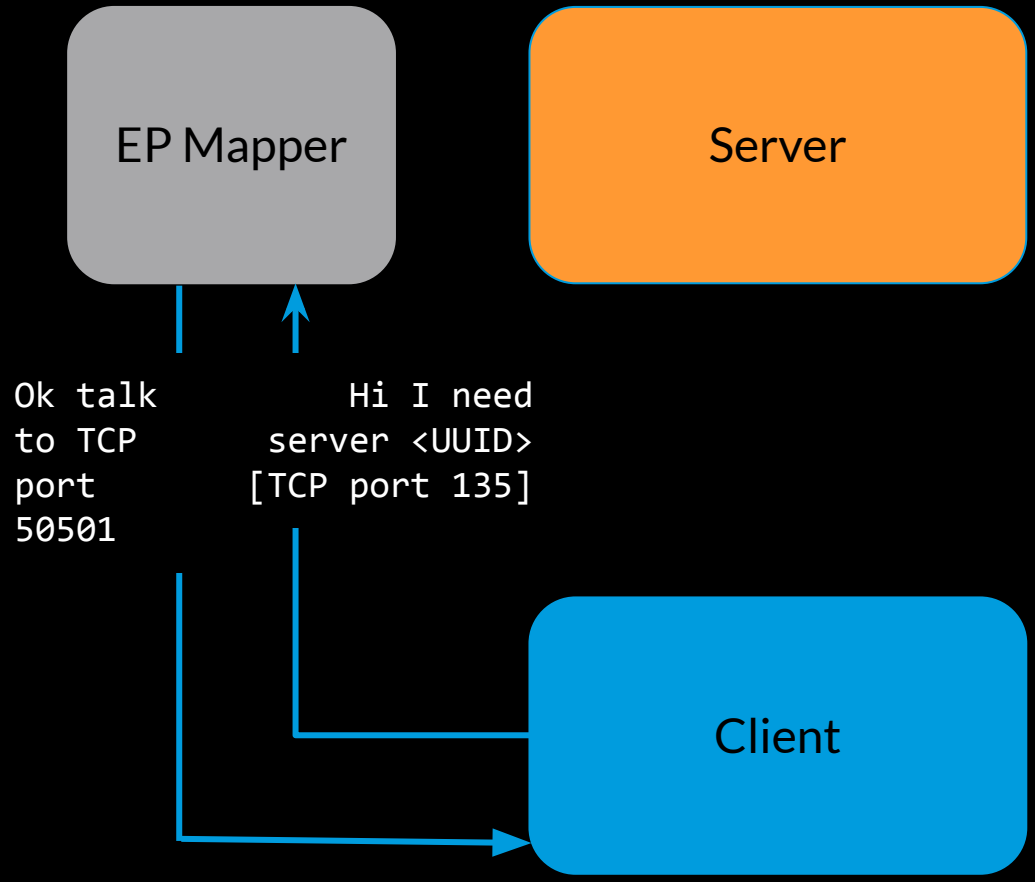


Akamai

Well-Known Endpoints

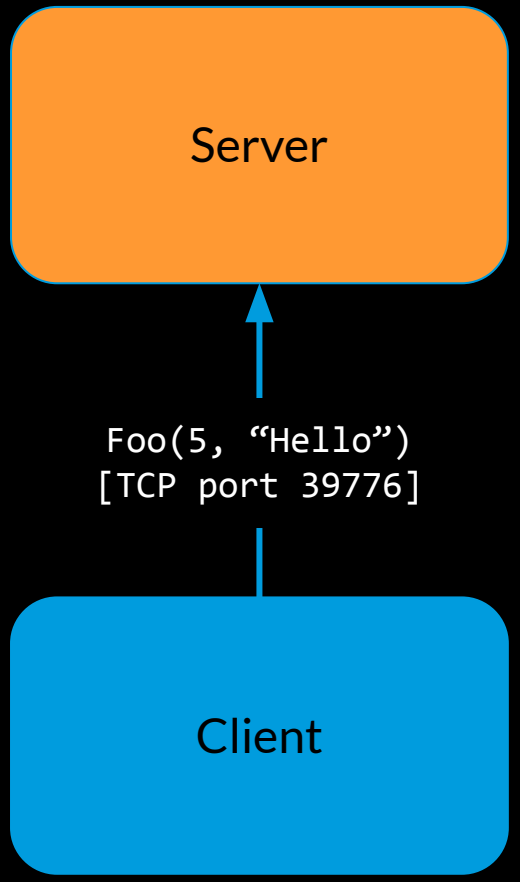


Dynamic Endpoints

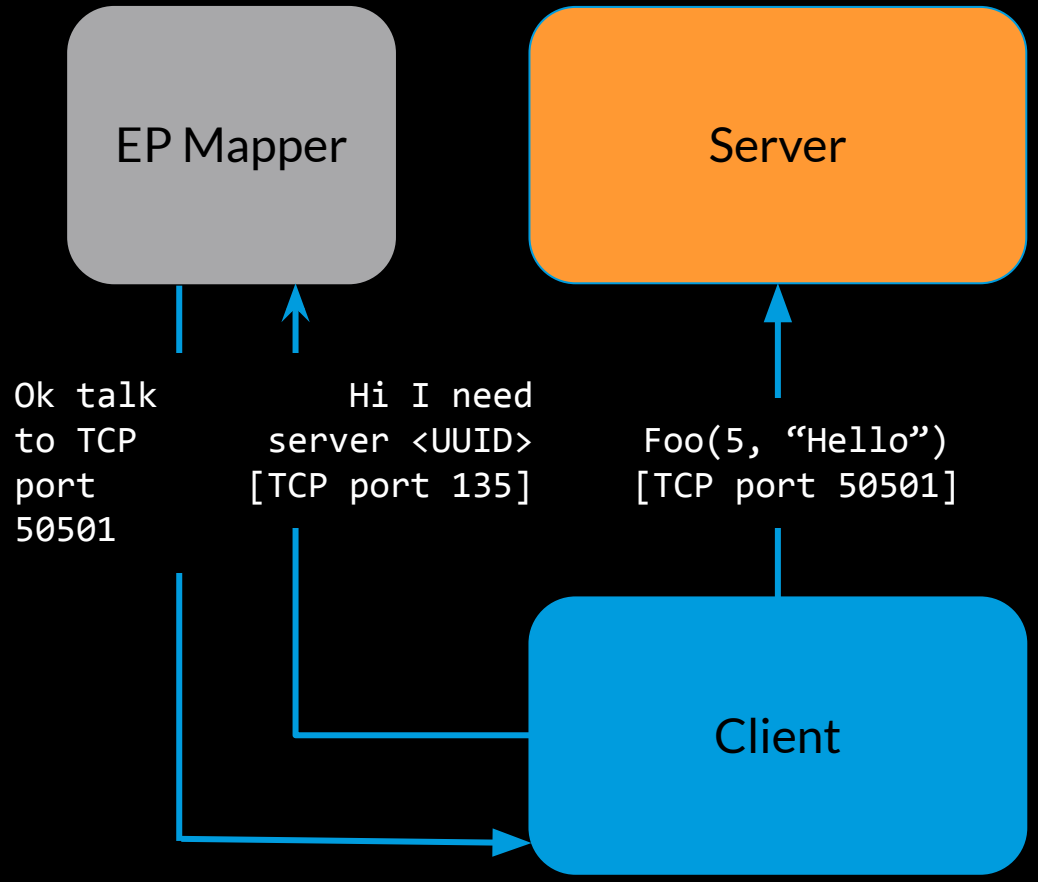


Akamai

Well-Known Endpoints



Dynamic Endpoints



Akamai

Name	Value	Purpose
GUID_ATSvc	1FF70682-0A51-30E8-076D-740BE8CEE98B	ATSvc UUID version 1.0
GUID_SASec	378E52B0-C0A9-11CF-822D-00AA0051E40F	SASec UUID version 1.0
GUID_ITaskSchedulerService	86D35949-83C9-4044-B424-DB363231FD0C	ITaskSchedulerService UUID version 1.0

Task Scheduler Service Remoting Protocol

Parameter	Value
RPC interface UUID	{367ABB81-9844-35F1-AD32-98F038001003}
Named pipe	\PIPE\svcctl

Service control manager remote protocol

Parameter	Value
RPC Well-Known Endpoint	\pipe\lsarpc<3>
RPC Interface UUID	{c681d488-d850-11d0-8c52-00c04fd90f7e}
RPC Well-Known Endpoint	\pipe\efsrpc
RPC Interface UUID	{df1941c5-fe89-4e79-bf10-463657acf44d}

Encrypting File System Remote (EFSRPC) Protocol

Task Scheduler Endpoint Resolution

172.17.0.61	172.17.0.20	TCP	66 63325 → 135 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0 MS
172.17.0.20	172.17.0.61	TCP	66 135 → 63325 [SYN, ACK, ECN] Seq=0 Ack=1 Win=8192 Le
172.17.0.61	172.17.0.20	TCP	54 63325 → 135 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
172.17.0.61	172.17.0.20	DCERPC	214 Bind: call_id: 2, Fragment: Single, 3 context items
172.17.0.20	172.17.0.61	DCERPC	162 Bind_ack: call_id: 2, Fragment: Single, max_xmit: 5
172.17.0.61	172.17.0.20	EPM	222 Map request, TaskSchedulerService, 32bit NDR
172.17.0.20	172.17.0.61	EPM	226 Map response, TaskSchedulerService, 32bit NDR
172.17.0.61	172.17.0.20	TCP	66 63326 → 49666 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0
172.17.0.20	172.17.0.61	TCP	66 49666 → 63326 [SYN, ACK, ECN] Seq=0 Ack=1 Win=8192
172.17.0.61	172.17.0.20	TCP	54 63326 → 49666 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
172.17.0.61	172.17.0.20	DCERPC	262 Bind: call_id: 2, Fragment: Single, 3 context items
172.17.0.20	172.17.0.61	DCERPC	388 Bind_ack: call_id: 2, Fragment: Single, max_xmit: 5
172.17.0.61	172.17.0.20	DCERPC	594 AUTH3: call_id: 2, Fragment: Single, NTLMSSP_AUTH,

Task Scheduler Endpoint Resolution

172.17.0.61	172.17.0.20	TCP	66 63325 → 135 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0 MSS
172.17.0.20	172.17.0.61	TCP	66 135 → 63325 [SYN, ACK, ECN] Seq=0 Ack=1 Win=8192 Len
172.17.0.61	172.17.0.20	TCP	54 63325 → 135 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
172.17.0.61	172.17.0.20	DCERPC	214 Bind: call_id: 2, Fragment: Single, 3 context items
172.17.0.20	172.17.0.61	DCERPC	162 Bind ack: call id: 2, Fragment: Single, max xmit: 5
172.17.0.61	172.17.0.20	EPM	222 Map request, TaskSchedulerService, 32bit NDR
172.17.0.20	172.17.0.61	EPM	226 Map response, TaskSchedulerService, 32bit NDR
172.17.0.61	172.17.0.20	TCP	66 63326 → 49666 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0
172.17.0.20	172.17.0.61	TCP	66 49666 → 63326 [SYN, ACK, ECN] Seq=0 Ack=1 Win=8192
172.17.0.61	172.17.0.20	TCP	54 63326 → 49666 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
172.17.0.61	172.17.0.20	DCERPC	262 Bind: call_id: 2, Fragment: Single, 3 context items
172.17.0.20	172.17.0.61	DCERPC	388 Bind_ack: call_id: 2, Fragment: Single, max_xmit: 5
172.17.0.61	172.17.0.20	DCERPC	594 AUTH3: call_id: 2, Fragment: Single, NTLMSSP_AUTH,

Task Scheduler Endpoint Resolution

172.17.0.61	172.17.0.20	TCP	66 63325 → 135 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0 MSS
172.17.0.20	172.17.0.61	TCP	66 135 → 63325 [SYN, ACK, ECN] Seq=0 Ack=1 Win=8192 Len
172.17.0.61	172.17.0.20	TCP	54 63325 → 135 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
172.17.0.61	172.17.0.20	DCERPC	214 Bind: call_id: 2, Fragment: Single, 3 context items
172.17.0.20	172.17.0.61	DCERPC	162 Bind ack: call id: 2, Fragment: Single, max xmit: 58
172.17.0.61	172.17.0.20	EPM	222 Map request, TaskSchedulerService, 32bit NDR
172.17.0.20	172.17.0.61	EPM	226 Map response, TaskSchedulerService, 32bit NDR
172.17.0.61	172.17.0.20	TCP	66 63326 → 49666 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0
172.17.0.20	172.17.0.61	TCP	66 49666 → 63326 [SYN, ACK, ECN] Seq=0 Ack=1 Win=8192
172.17.0.61	172.17.0.20	TCP	54 63326 → 49666 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
172.17.0.61	172.17.0.20	DCERPC	262 Bind: call_id: 2, Fragment: Single, 3 context items
172.17.0.20	172.17.0.61	DCERPC	388 Bind_ack: call_id: 2, Fragment: Single, max_xmit: 58
172.17.0.61	172.17.0.20	DCERPC	594 AUTH3: call_id: 2, Fragment: Single, NTLMSSP_AUTH,

Task Scheduler Endpoint Resolution

172.17.0.61	172.17.0.20	TCP	66 63325 → 135 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0 MS
172.17.0.20	172.17.0.61	TCP	66 135 → 63325 [SYN, ACK, ECN] Seq=0 Ack=1 Win=8192 Le
172.17.0.61	172.17.0.20	TCP	54 63325 → 135 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
172.17.0.61	172.17.0.20	DCERPC	214 Bind: call_id: 2, Fragment: Single, 3 context items
172.17.0.20	172.17.0.61	DCERPC	162 Bind ack: call id: 2, Fragment: Single, max xmit: 5
		EPM	222 Map request, TaskSchedulerService, 32bit NDR
		EPM	226 Map response, TaskSchedulerService, 32bit NDR
		TCP	66 63326 → 49666 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0
		TCP	66 49666 → 63326 [SYN, ACK, ECN] Seq=0 Ack=1 Win=8192
		TCP	54 63326 → 49666 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
		DCERPC	262 Bind: call_id: 2, Fragment: Single, 3 context items
		DCERPC	388 Bind_ack: call_id: 2, Fragment: Single, max_xmit: 5
		DCERPC	594 AUTH3: call_id: 2, Fragment: Single, NTLMSSP_AUTH,

```
DCE/RPC Endpoint Mapper, Map
  Operation: Map (3)
  [Request in frame: 1071]
  Handle: 00000000000000000000000000000000000000000000
  Num Towers: 1
▾ Tower array:
    Max Count: 4
    Offset: 0
    Actual Count: 1
  ▾ Tower pointer:
      Referent ID: 0x0000000000000003
      Length: 75
      Length: 75
      Number of floors: 5
    › Floor 1 UUID: TaskSchedulerService
    › Floor 2 UUID: 32bit NDR
    › Floor 3 RPC connection-oriented protocol
    › Floor 4 TCP Port:49666
    ▾ Floor 5 IP:172.17.0.20
```

Akamai

Task Scheduler Endpoint Resolution

172.17.0.61	172.17.0.20	TCP	66 63325 → 135 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0 MS
172.17.0.20	172.17.0.61	TCP	66 135 → 63325 [SYN, ACK, ECN] Seq=0 Ack=1 Win=8192 Le
172.17.0.61	172.17.0.20	TCP	54 63325 → 135 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
172.17.0.61	172.17.0.20	DCERPC	214 Bind: call_id: 2, Fragment: Single, 3 context items
172.17.0.20	172.17.0.61	DCERPC	162 Bind_ack: call_id: 2, Fragment: Single, max_xmit: 5
172.17.0.61	172.17.0.20	EPM	222 Map request, TaskSchedulerService, 32bit NDR
172.17.0.20	172.17.0.61	EPM	226 Map response, TaskSchedulerService, 32bit NDR
172.17.0.61	172.17.0.20	TCP	66 63326 → 49666 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0
172.17.0.20	172.17.0.61	TCP	66 49666 → 63326 [SYN, ACK, ECN] Seq=0 Ack=1 Win=8192
172.17.0.61	172.17.0.20	TCP	54 63326 → 49666 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
172.17.0.61	172.17.0.20	DCERPC	262 Bind: call_id: 2, Fragment: Single, 3 context items
172.17.0.20	172.17.0.61	DCERPC	388 Bind_ack: call_id: 2, Fragment: Single, max_xmit: 5
172.17.0.61	172.17.0.20	DCERPC	594 AUTH3: call_id: 2, Fragment: Single, NTLMSSP_AUTH,

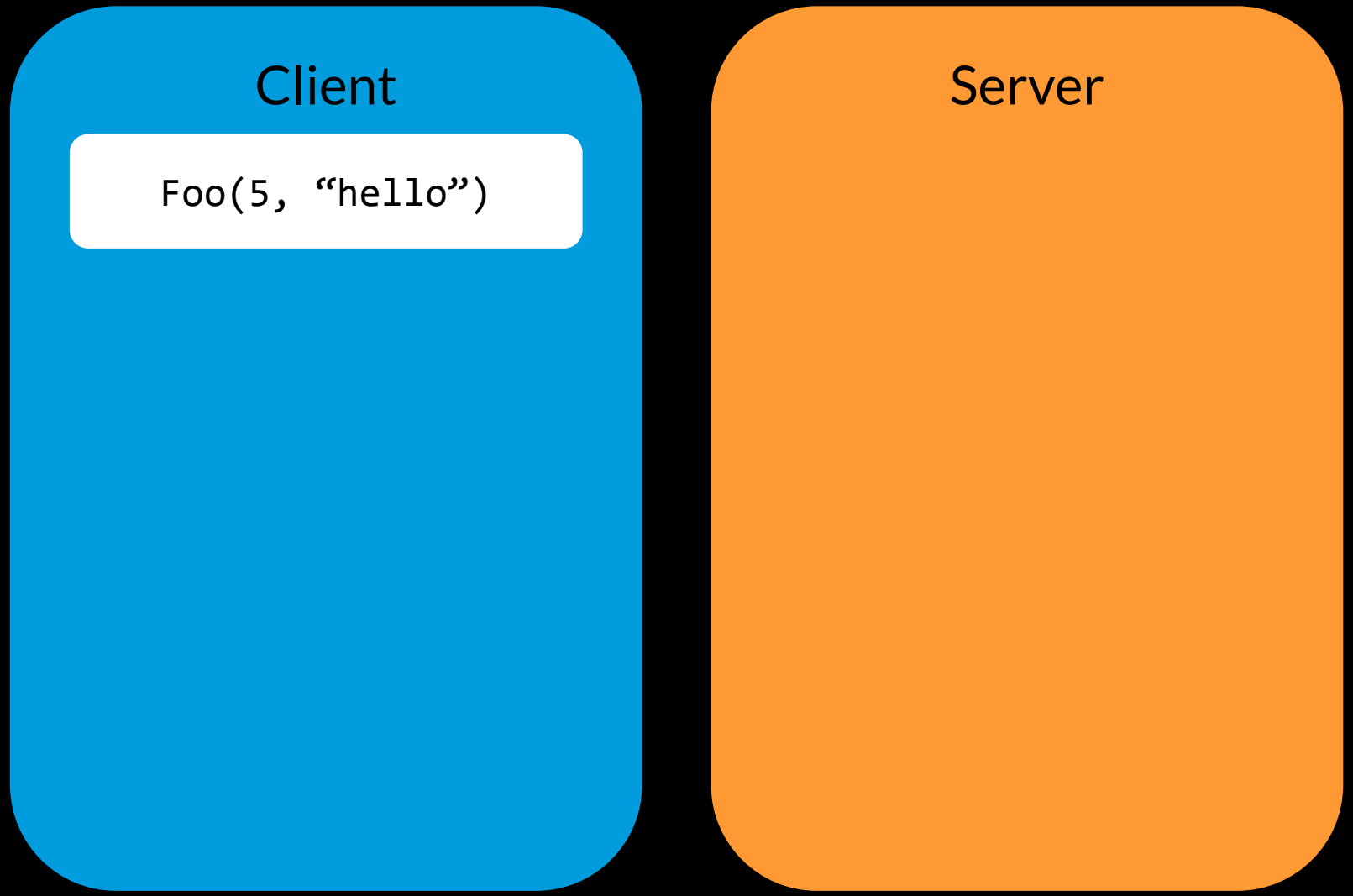
Task Scheduler Endpoint Resolution

172.17.0.61	172.17.0.20	TCP	66 63325 → 135 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0 MSS
172.17.0.20	172.17.0.61	TCP	66 135 → 63325 [SYN, ACK, ECN] Seq=0 Ack=1 Win=8192 Len
172.17.0.61	172.17.0.20	TCP	54 63325 → 135 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
172.17.0.61	172.17.0.20	DCERPC	214 Bind: call_id: 2, Fragment: Single, 3 context items
172.17.0.20	172.17.0.61	DCERPC	162 Bind_ack: call_id: 2, Fragment: Single, max_xmit: 58
172.17.0.61	172.17.0.20	EPM	222 Map request, TaskSchedulerService, 32bit NDR
172.17.0.20	172.17.0.61	EPM	226 Map response, TaskSchedulerService, 32bit NDR
172.17.0.61	172.17.0.20	TCP	66 63326 → 49666 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0 M
172.17.0.20	172.17.0.61	TCP	66 49666 → 63326 [SYN, ACK, ECN] Seq=0 Ack=1 Win=8192 L
172.17.0.61	172.17.0.20	TCP	54 63326 → 49666 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
172.17.0.61	172.17.0.20	DCERPC	262 Bind: call_id: 2, Fragment: Single, 3 context items
172.17.0.20	172.17.0.61	DCERPC	388 Bind_ack: call_id: 2, Fragment: Single, max_xmit: 5
172.17.0.61	172.17.0.20	DCERPC	594 AUTH3: call_id: 2, Fragment: Single, NTLMSSP_AUTH,

Binding

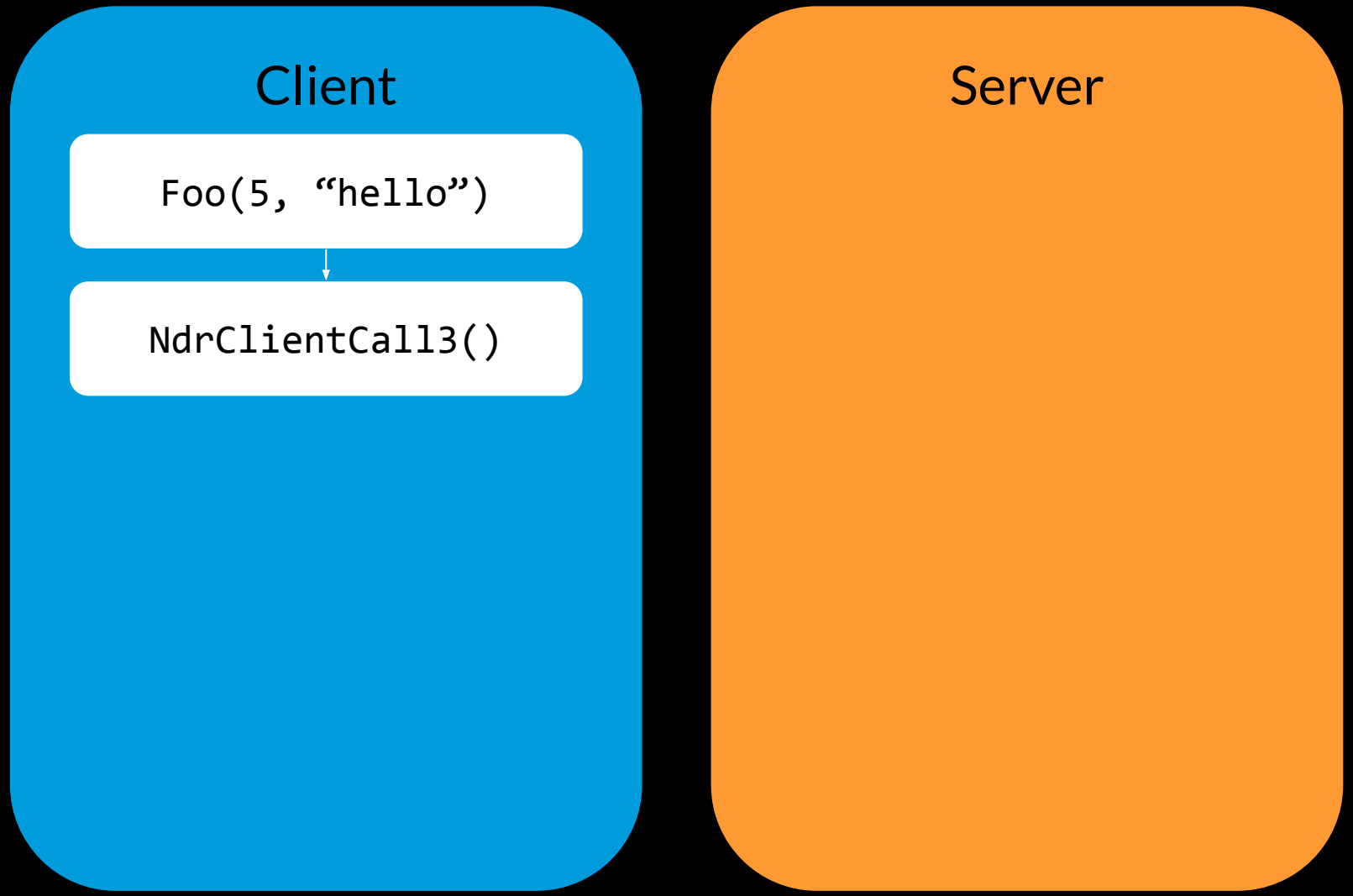
- The representation of a session between a client and a server
  - Practically, a handle
  - Client and server can manipulate binding data using designated functions
  - Used for authentication (among other things)

An RPC Call's Flow



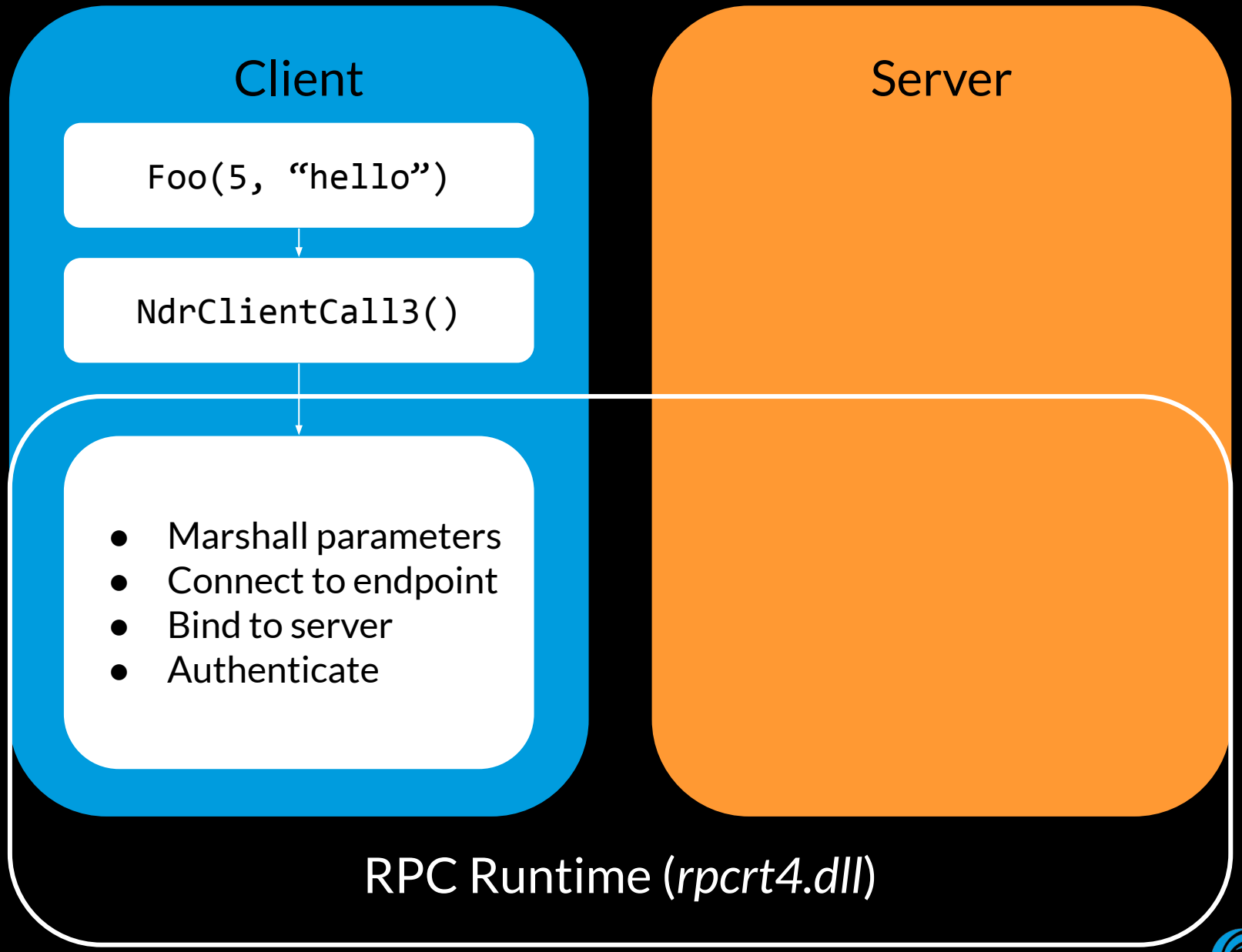
Akamai

An RPC Call's Flow



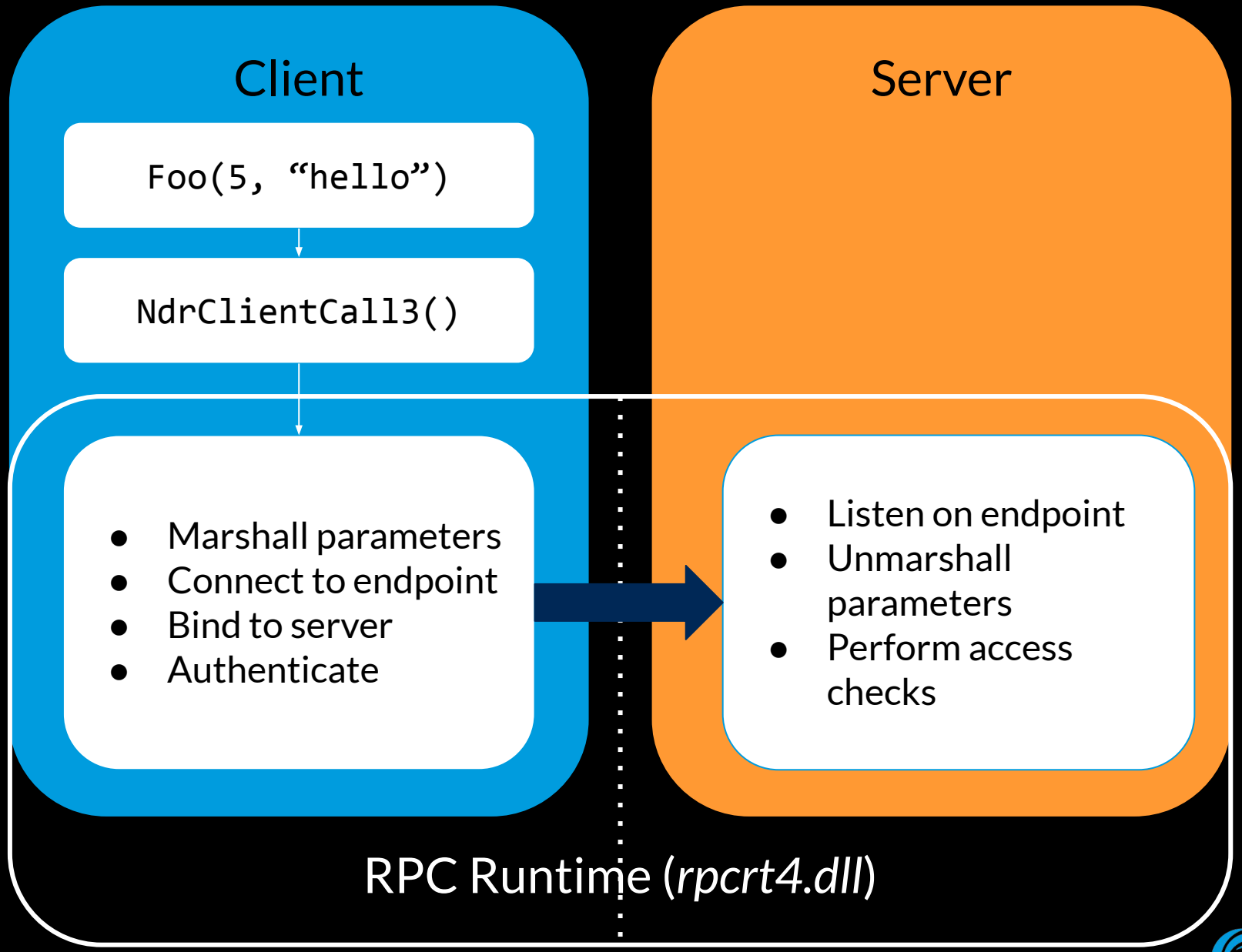
Akamai

An RPC Call's Flow

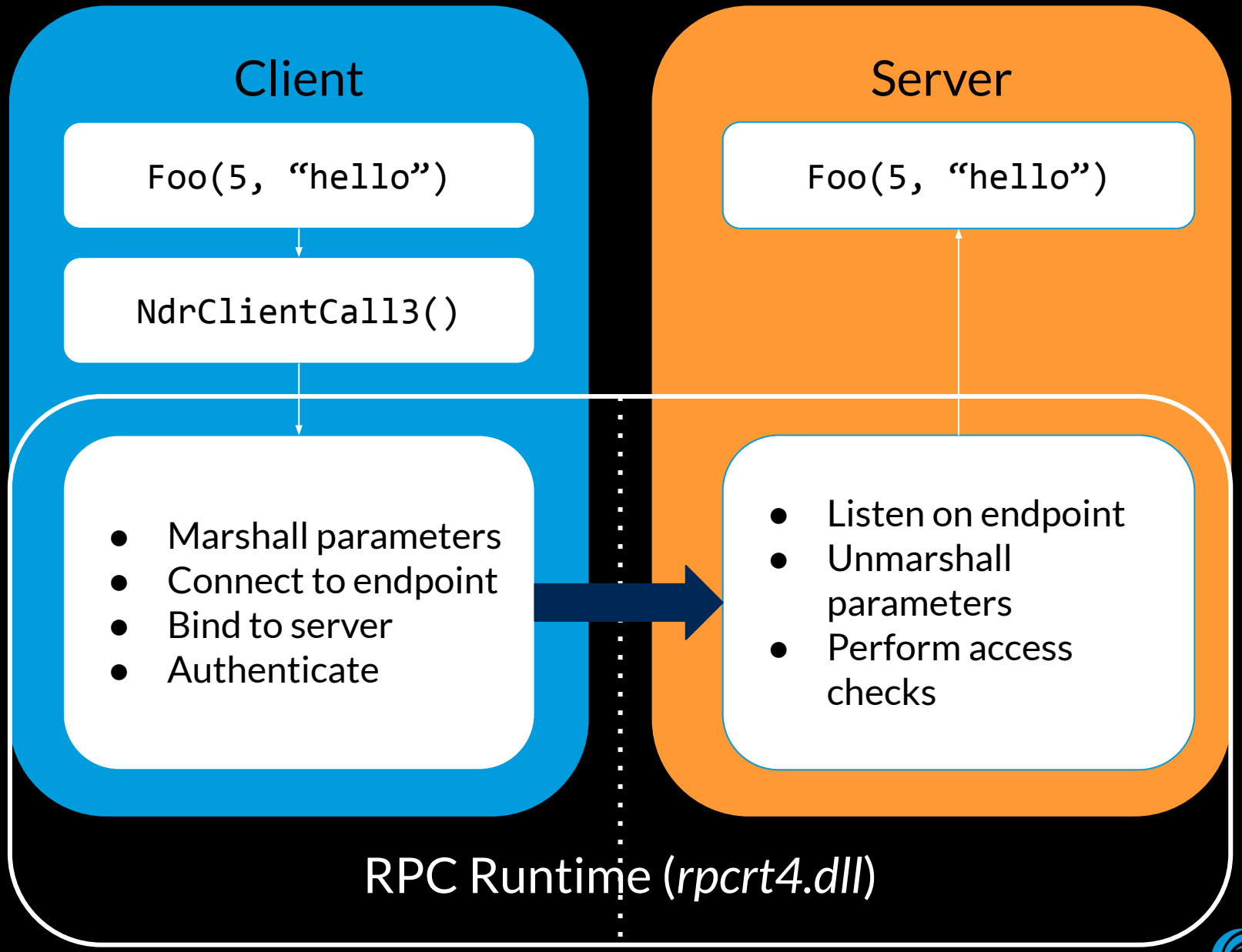


Akamai

An RPC Call's Flow



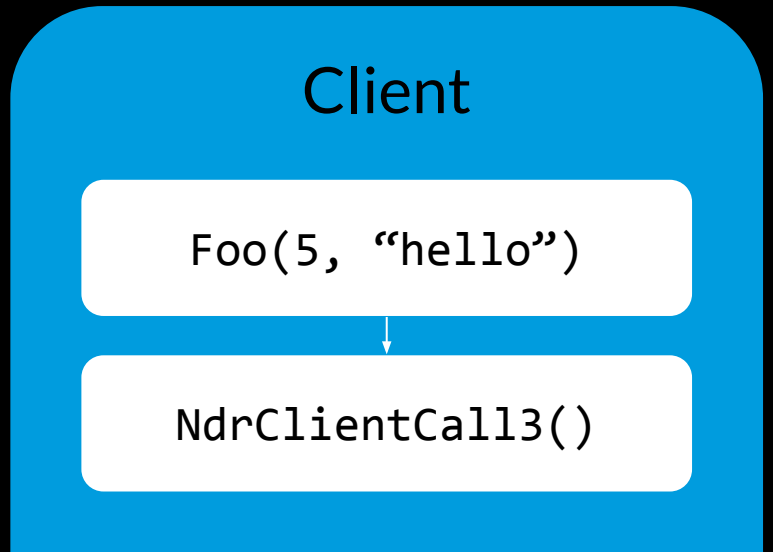
An RPC Call's Flow



Zooming In

IDL:

```
void Foo([in] int number,
         [in] char* message);
```



Akamai

Zooming In

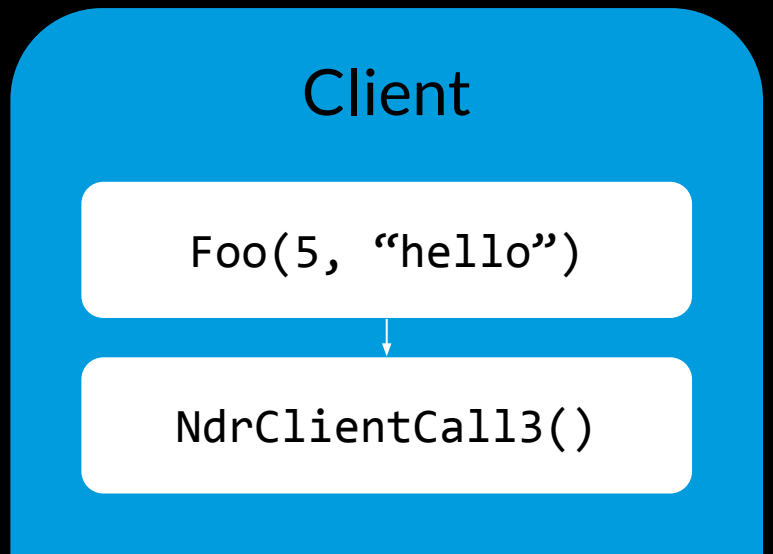
IDL:

```
void Foo([in] int number,
         [in] char* message);
```

Test_c.c:

```
void Foo(
  handle_t IDL_handle,
  int number,
  unsigned char *message) {

  NdrClientCall3(
  (PMIDL_STUBLESS_PROXY_INFO
  )&Test_ProxyInfo, 0, 0,
  IDL_handle, number, message);
}
```



Akamai

Zooming In

IDL:

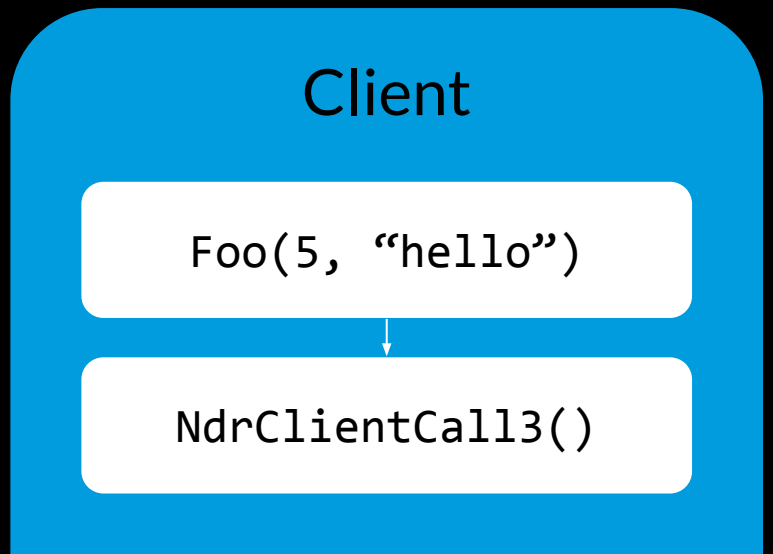
```
void Foo([in] int number,
         [in] char* message);
```

Test_c.c:

```
void Foo(
  handle_t IDL_handle,
  int number,
  unsigned char *message) {

  NdrClientCall3(
  (PMIDL_STUBLESS_PROXY_INFO
  )&Test_ProxyInfo, 0, 0,
  IDL_handle, number, message);
}
```

Opnum



Akamai

Quick Recap

- Interface – describes server functionality [UUID]
- Transport – the communication medium [protocol sequence]
- Endpoint – destination to connect to [port, pipe name, etc.]
- Binding – represents a client-server session [binding handle]

MS-RPC (In-)Security

Agenda for this part

- MS-RPC built-in security mechanisms
- Security-related problems in MS-RPC

Security Mechanisms

- It’s a complete mess
- We’ll focus on **remote communication** and cover:
  - Authentication
  - Security descriptors
  - Security callback

Flags (🏳) – specified during interface registration

```
RPC_STATUS RpcServerRegisterIf3(
  RPC_IF_HANDLE IfSpec,
  UUID          *MgrTypeUuid,
  RPC_MGR_EPV   *MgrEpv
  unsigned int  Flags,
  unsigned int  MaxCalls,
  unsigned int  MaxRpcSize,
  RPC_IF_CALLBACK_FN *IfCallback,
  void          *SecurityDescriptor
);
```

Transport Layer Authentication

SMB Authentication

- Named pipes are carried over SMB, requesting IPC$ share

SMB Authentication

- Named pipes are carried over SMB, requesting IPC$ share
- Authentication is on the SMB level

→ requires a valid user

SMB Authentication

- Named pipes are carried over SMB, requesting IPC$ share
- Authentication is on the SMB level

  → requires a valid user

- NULL sessions aren't supported anymore

  → unless against DC:

  \pipe*netlogon*, *pipe**samr*, *pipe**lsarpc*

Authenticated Binding

Authenticated Binding

- Binding that has authentication info

Authenticated Binding

- Binding that has authentication info
- Both server and client can set auth info using

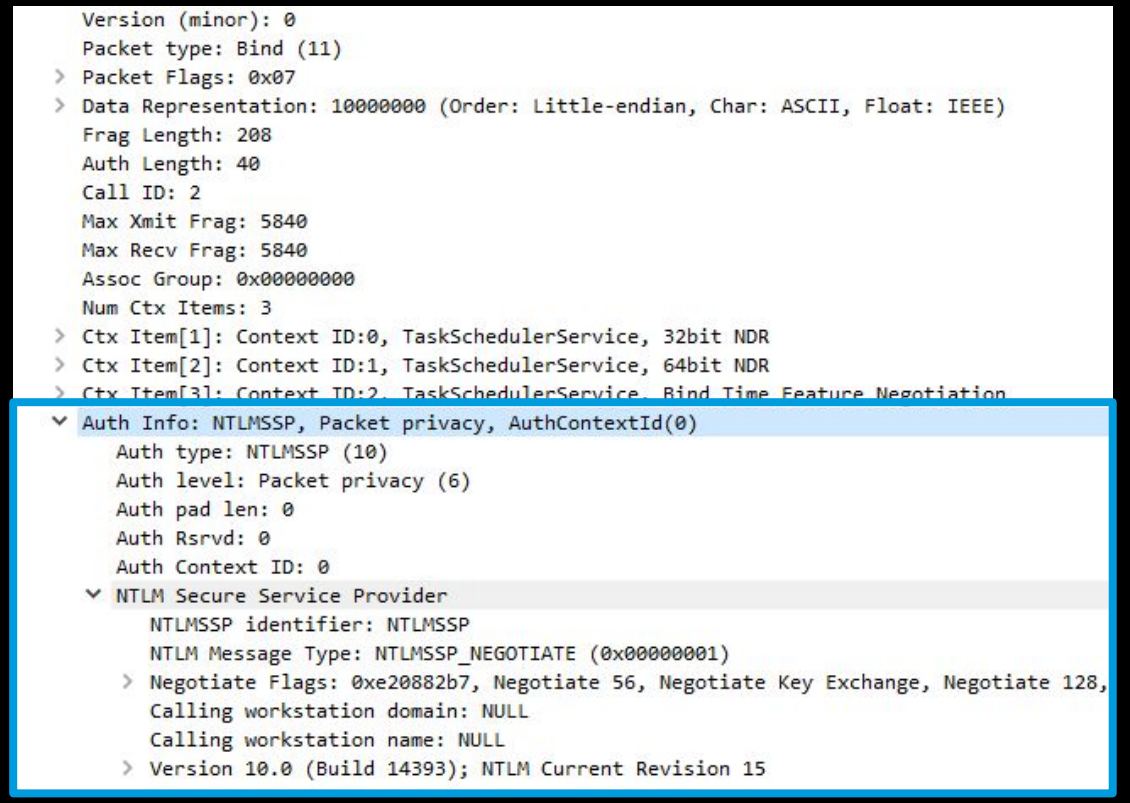
  `RpcServerRegisterAuthInfo`, `RpcBindingSetAuthInfo`

Authenticated Binding

- Binding that has authentication info
- Both server and client can set auth info using `RpcServerRegisterAuthInfo`, `RpcBindingSetAuthInfo`
- Provides identity-based access control and other protections (e.g, Replay prevention, Integrity, Confidentiality) - specified by authentication level

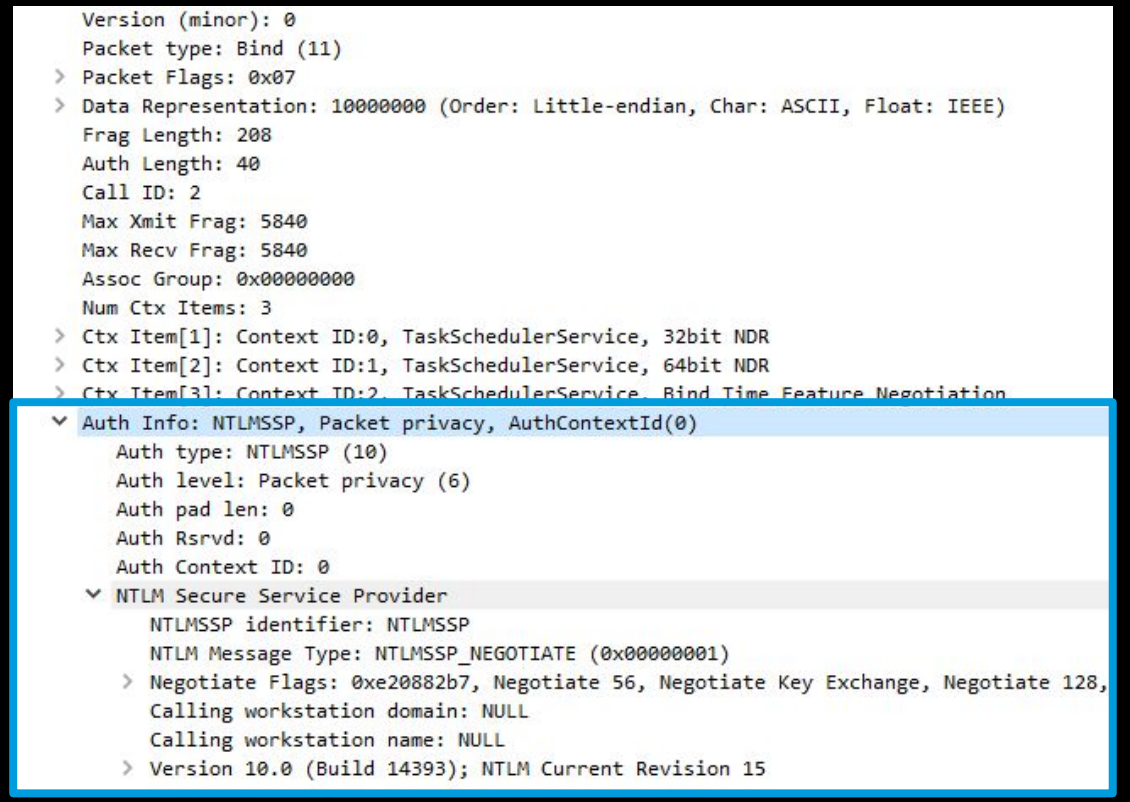
Authenticated Binding

- RPC client and server exchange bind/bind_ack messages with authentication information



Authenticated Binding

- RPC client and server exchange bind/bind_ack messages with authentication information
- End result: a security context - a “security binding”



Authenticated Binding

- The client isn’t forced authenticate, even if the server registered authentication!

Authenticated Binding

- The client isn't forced authenticate, even if the server registered authentication!

Client\Server	Unauthenticated Binding NoFlags, NoSecurityCallback	Unauthenticated Binding NoFlags, SecurityCallback	Unauthenticated Binding Flags[1], NoSecurityCallback	Unauthenticated Binding Flags[1], SecurityCallback	Authenticated Binding NoFlags, NoSecurityCallback
Unauthenticated Binding	Success	Error 5 (Access Denied)	Success	Success	Success

https://csandker.io/2021/02/21/Offensive-Windows-IPC-2-RPC.html

Authenticated Binding

🏳 `RPC_IF_ALLOW_SECURE_ONLY`

Security Descriptors

```
RPC_STATUS RpcServerRegisterIf3(
  RPC_IF_HANDLE IfSpec,
  UUID          *MgrTypeUuid,
  RPC_MGR_EPV   *MgrEpv
  unsigned int  Flags,
  unsigned int  MaxCalls,
  unsigned int  MaxRpcSize,
  RPC_IF_CALLBACK_FN *IfCallback,
  void          *SecurityDescriptor
);
```

Security Descriptors

- RPC servers can set security descriptors on both the endpoint and the interface

```
RPC_STATUS RpcServerRegisterIf3(
  RPC_IF_HANDLE IfSpec,
  UUID          *MgrTypeUuid,
  RPC_MGR_EPV   *MgrEpv
  unsigned int  Flags,
  unsigned int  MaxCalls,
  unsigned int  MaxRpcSize,
  RPC_IF_CALLBACK_FN *IfCallback,
  void          *SecurityDescriptor
);
```

Security Descriptors

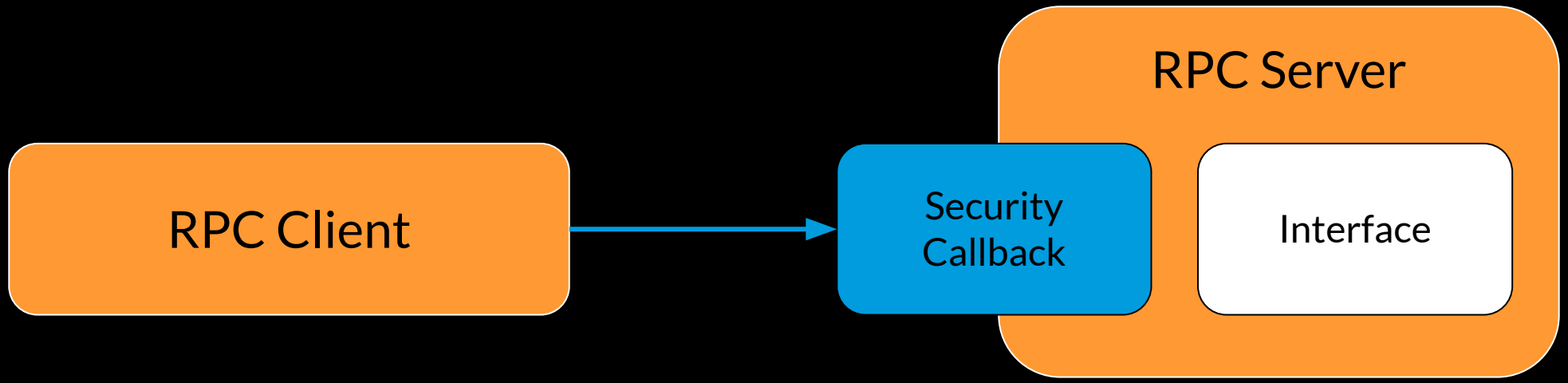
```
PS C:\Users\defcon> ConvertFrom-SddlString
"D:(A;;GR;;;AN)(A;;GR;;;WD)(A;;GR;;;RC)(A;;GA;;;BA)(A;;GA;;;SY)(A;;GR;;;AC)(A;;GR;;;S-1-15-2-2)"

Owner              :
Group              :
DiscretionaryAcl : {Everyone: AccessAllowed (GenericRead), NT AUTHORITY\ANONYMOUS LOGON:
                   AccessAllowed (GenericRead), NT AUTHORITY\RESTRICTED: AccessAllowed
                   (GenericRead), NT AUTHORITY\SYSTEM: AccessAllowed (GenericAll)...}
SystemAcl          : {}
RawDescriptor      : System.Security.AccessControl.CommonSecurityDescriptor
```

appidsvc.dll

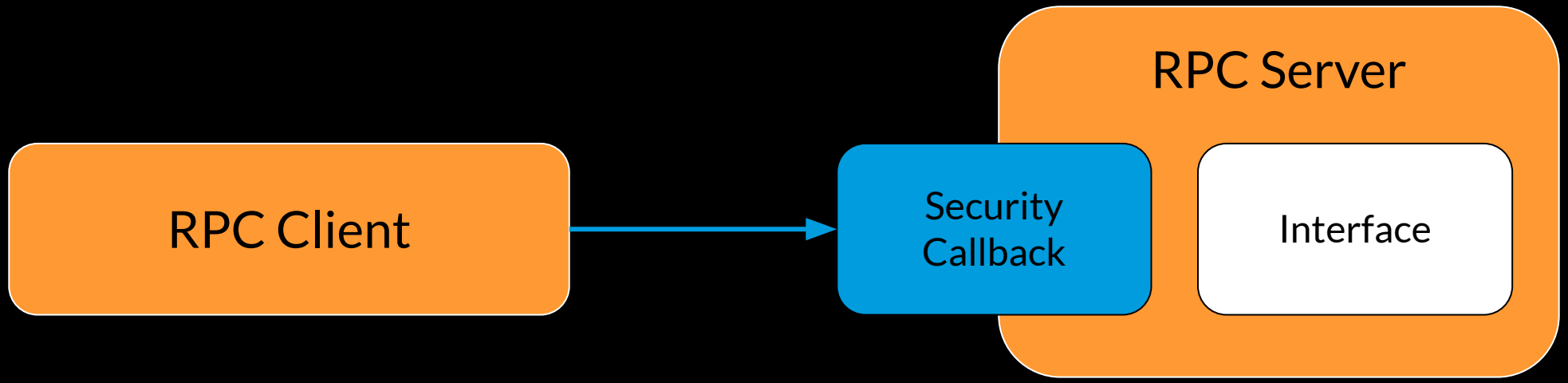
```
RPC_STATUS RpcServerRegisterIf3(
  RPC_IF_HANDLE IfSpec,
  UUID          *MgrTypeUuid,
  RPC_MGR_EPV   *MgrEpv
  unsigned int  Flags,
  unsigned int  MaxCalls,
  unsigned int  MaxRpcSize,
  RPC_IF_CALLBACK_FN *IfCallback,
  void          *SecurityDescriptor
);
```

Security Callback



Security Callback

```
RPC_STATUS RpcIfCallbackFn(
        RPC_IF_HANDLE InterfaceUuid,
        void *Context
)
{...}
```



Akamai

Task Scheduler

```
RPC_STATUS RpcServer::SecurityCallback(RPC_IF_HANDLE InterfaceUuid, void *Context) {
  …
  Status = RpcServerInqCallAttributesW(Context, &RpcCallAttributes);
  if ( !Status && RpcCallAttributes.AuthenticationLevel >=
RPC_C_AUTHN_LEVEL_PKT_PRIVACY ) {
    if ( RpcCallAttributes.ProtocolSequence == RPC_PROTSEQ_LRPC ) {
      return RPC_S_OK;
    }
    else if ( UuidEqual(&RpcCallAttributes.InterfaceUuid, &GUID_ITaskSchedulerService,
&Status) && !Status ) {
      …
    }
  }
  return RPC_S_ACCESS_DENIED;
}
```

IAS (Internet Authentication Service)

```
RPC_STATUS CIasRpcServer::RpcIfSecurityCallback(RPC_IF_HANDLE InterfaceUuid, void
*Context) {
  …
  if ( !I_RpcBindingIsClientLocal(0i64, &ClientLocalFlag) && ClientLocalFlag ) {
    if ( !RpcBindingInqAuthClientW(Context, 0i64, 0i64, &AuthnLevel, 0i64, 0i64)
      && AuthnLevel >= RPC_C_AUTHN_LEVEL_PKT_PRIVACY
      && CIasRpcServer::IsCorrectProtseq(&hBinding)
      && CIasRpcServer::IsAccessGranted(v3, &hBinding) )
    {
      return RPC_S_OK;
    }
  }
  return RPC_S_ACCESS_DENIED;
}
```

DHCP

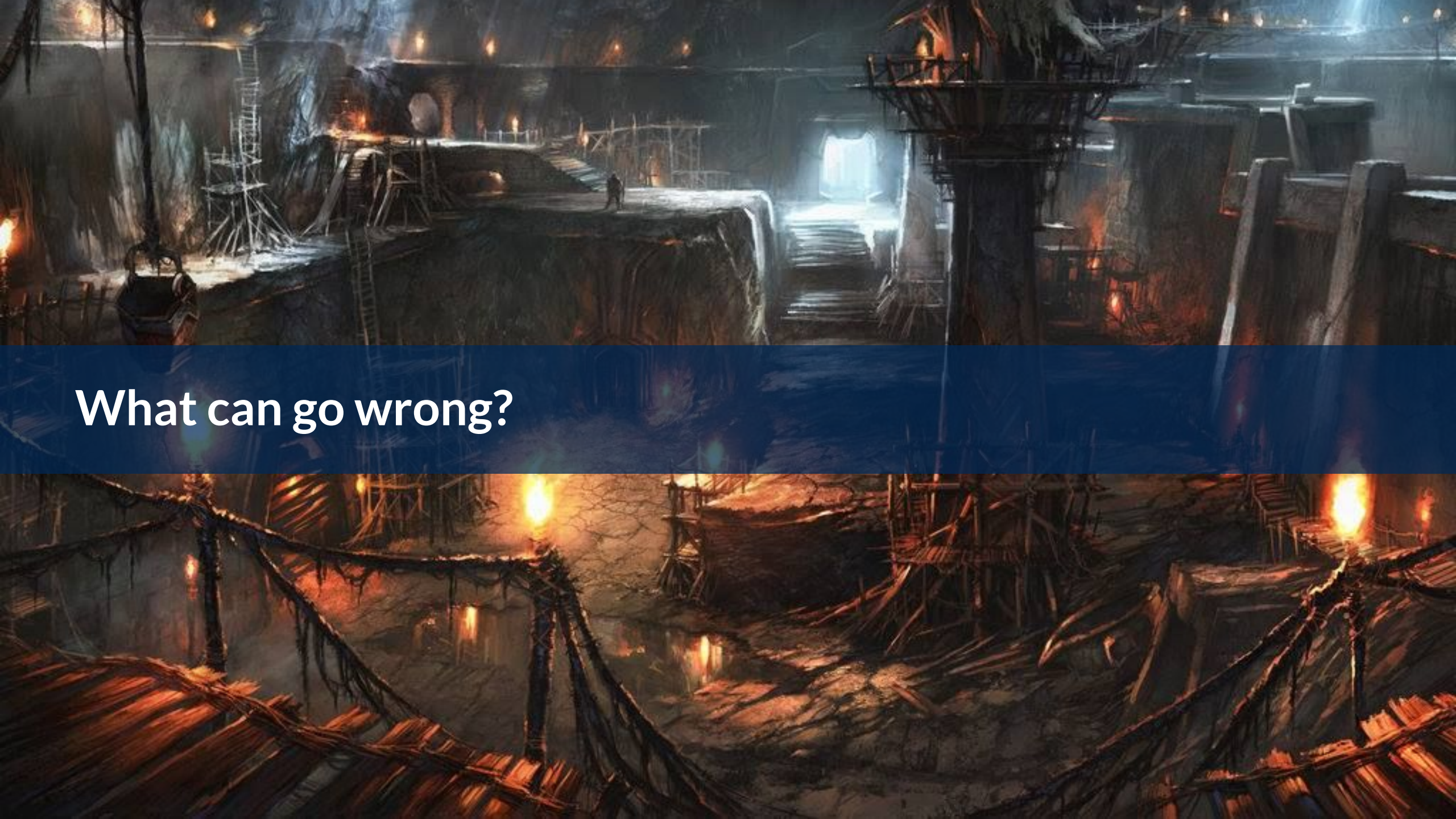
```
RPC_STATUS DhcpRpcCallback(RPC_IF_HANDLE InterfaceUuid, void *Context) {
  shouldPass = 0;
  if ( !RpcBindingToStringBindingW(Context, &StringBinding)
    && !RpcStringBindingParseW(StringBinding, 0i64, &Protseq, 0i64, 0i64, 0i64)
    && !_wcsicmp(Protseq, L"ncalrpc") ) {
    shouldPass = 1;
  }
  if ( Protseq ) RpcStringFreeW(&Protseq);
  if ( StringBinding ) RpcStringFreeW(&StringBinding);
  if ( shouldPass ) return RPC_S_OK;
  else
    return RPC_S_ACCESS_DENIED;
}
```

LSASS

```
RPC_STATUS LsaRpcIfCallbackFn(RPC_IF_HANDLE InterfaceUuid, void *Context)) {
  …
  LastError = RpcServerInqCallAttributesW(a2, &RpcCallAttributes);
  …
  if ( RpcCallAttributes.OpNum >= 0x86u ) return RPC_S_PROCNUM_OUT_OF_RANGE;
  …
  v6 = *((_DWORD *)&LsapRPCFunctionProperties + 2 * RpcCallAttributes.OpNum);
  if ( !_bittest(&v6, RpcCallAttributes.ProtocolSequence) )
    return RPC_S_PROTSEQ_NOT_SUPPORTED;
  …
}
```

Relevant Flags

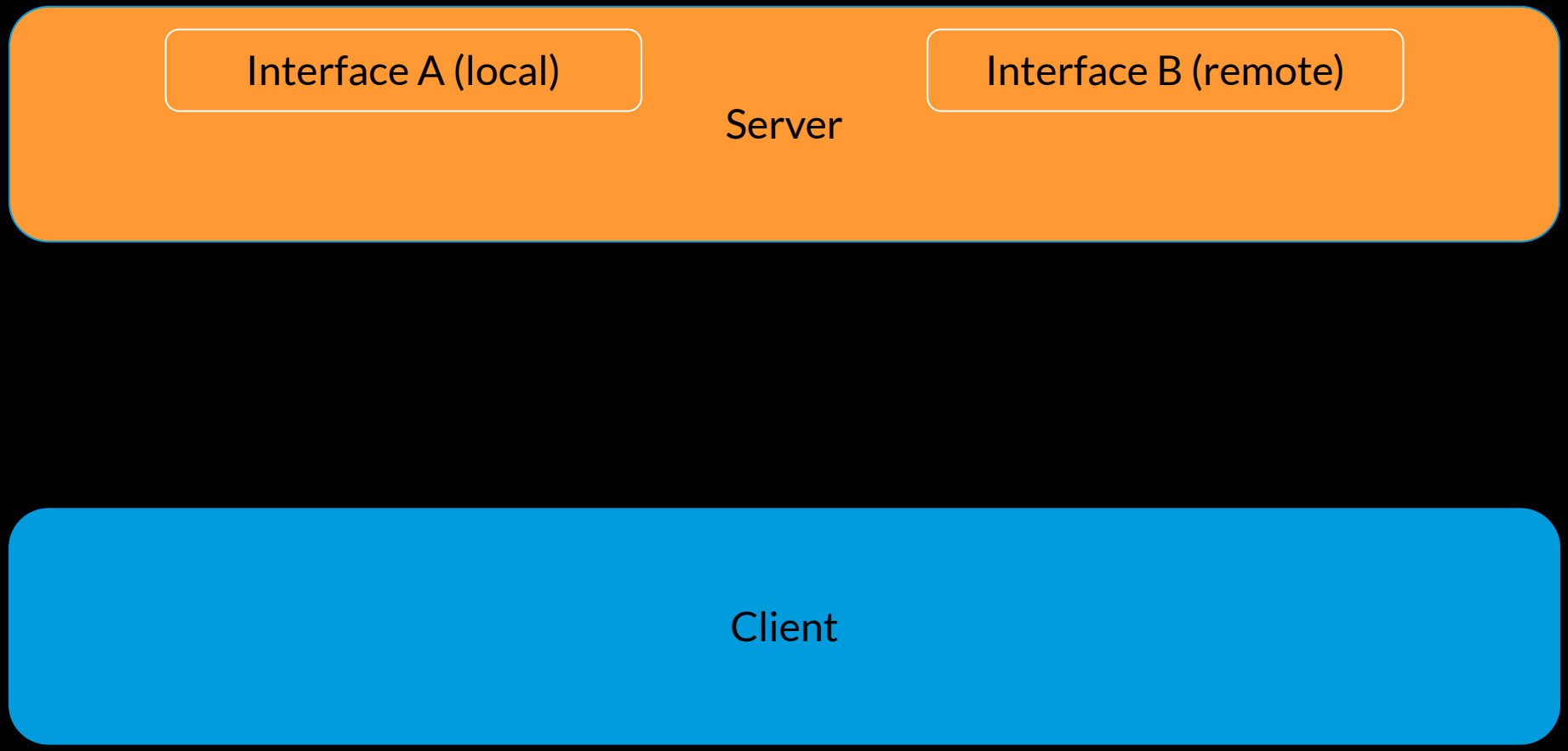
🏳 `RPC_IF_ALLOW_CALLBACKS_WITH_NO_AUTH`



What can go wrong?

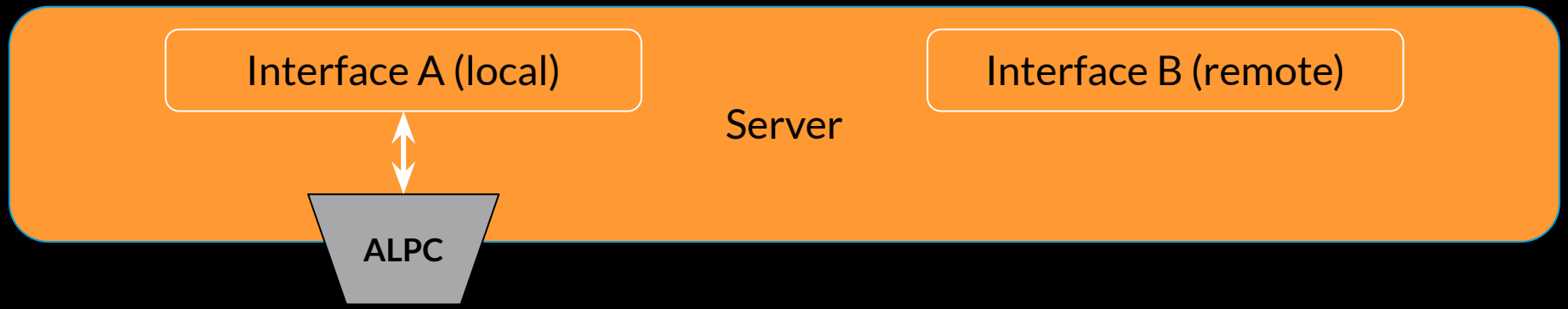
https://docs.microsoft.com/en-us/windows/win32/rpc/be-wary-of-other-rpc-endpoints-running-in-the-same-process

“Endpoint Multiplexing”

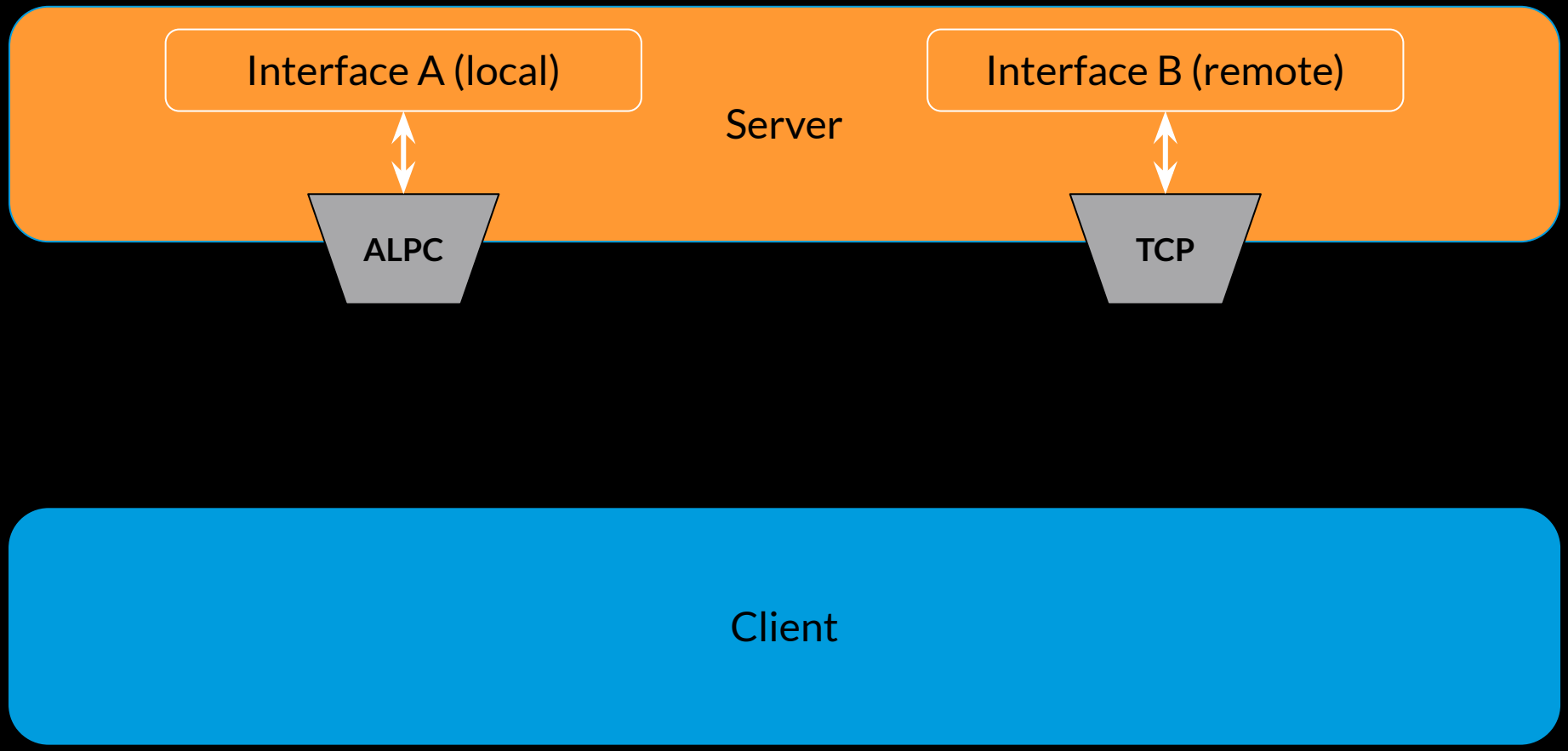


Akamai

“Endpoint Multiplexing”

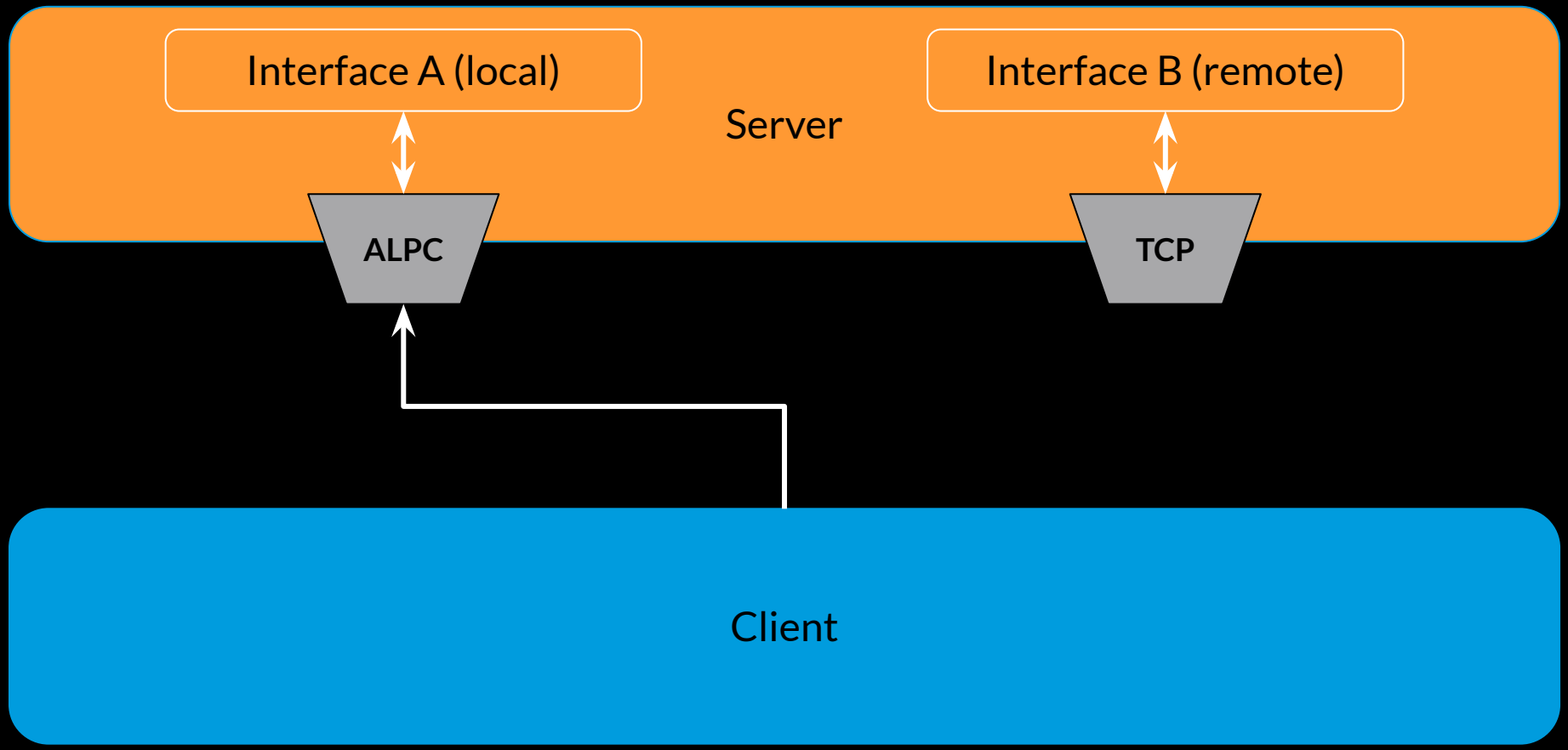


“Endpoint Multiplexing”



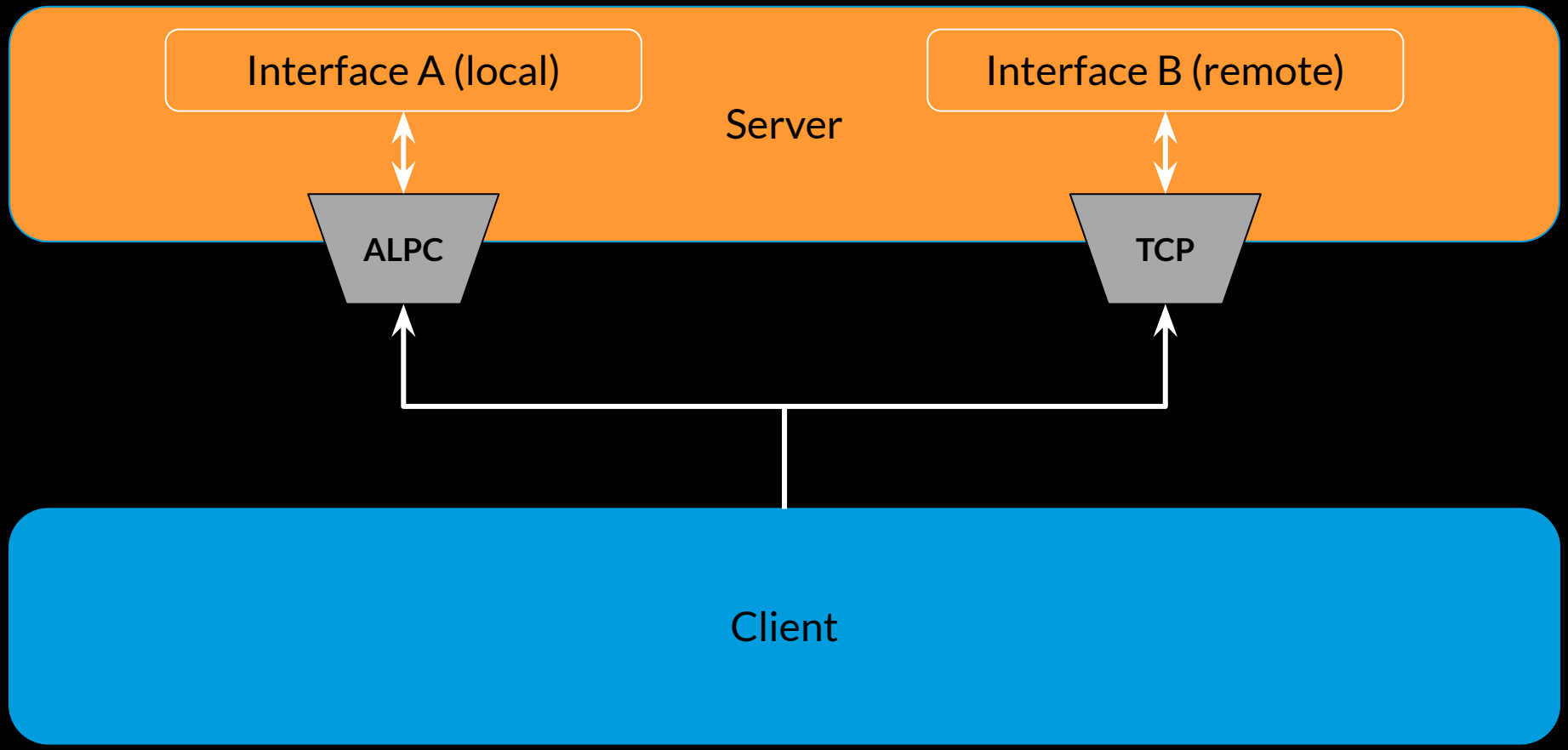
Akamai

“Endpoint Multiplexing”



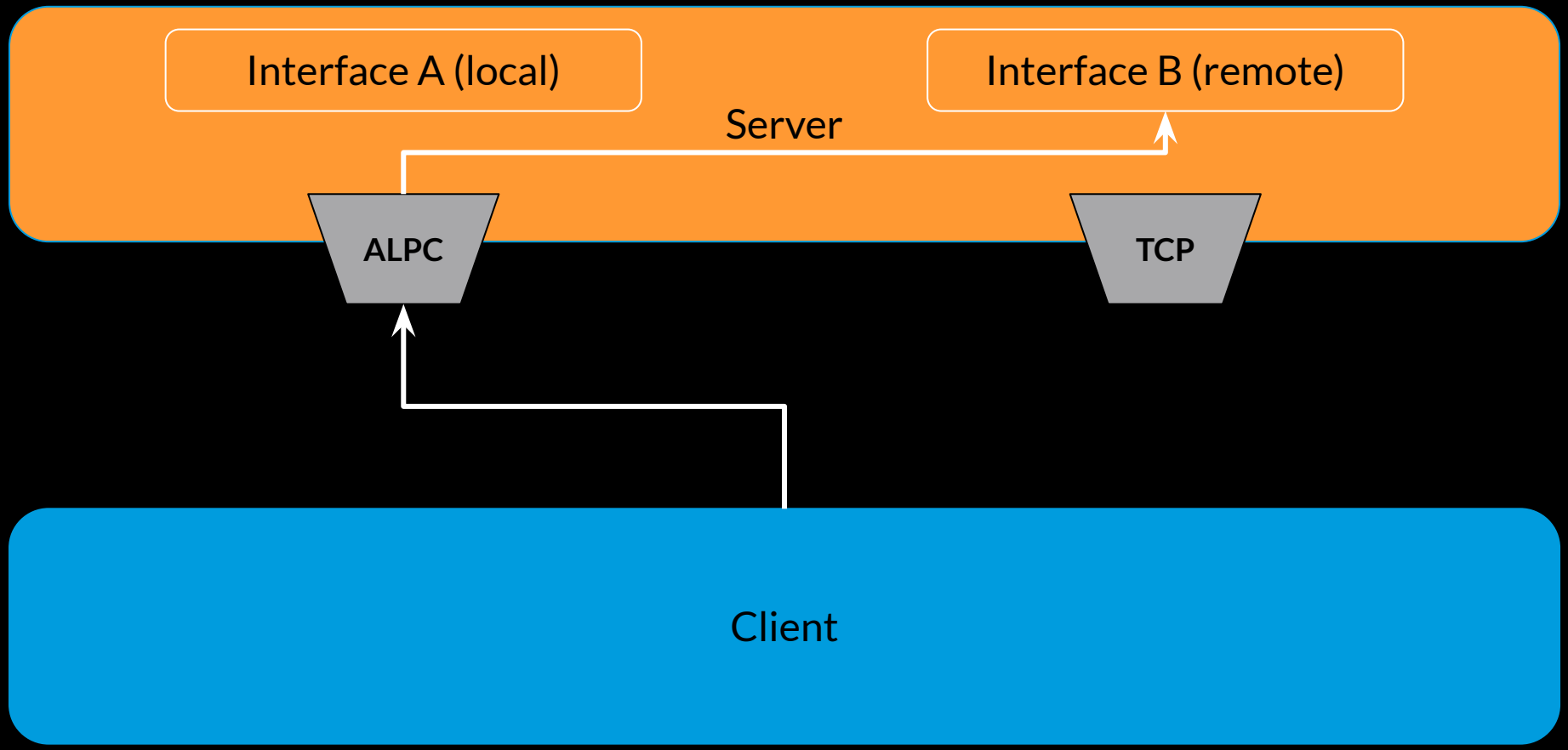
Akamai

“Endpoint Multiplexing”



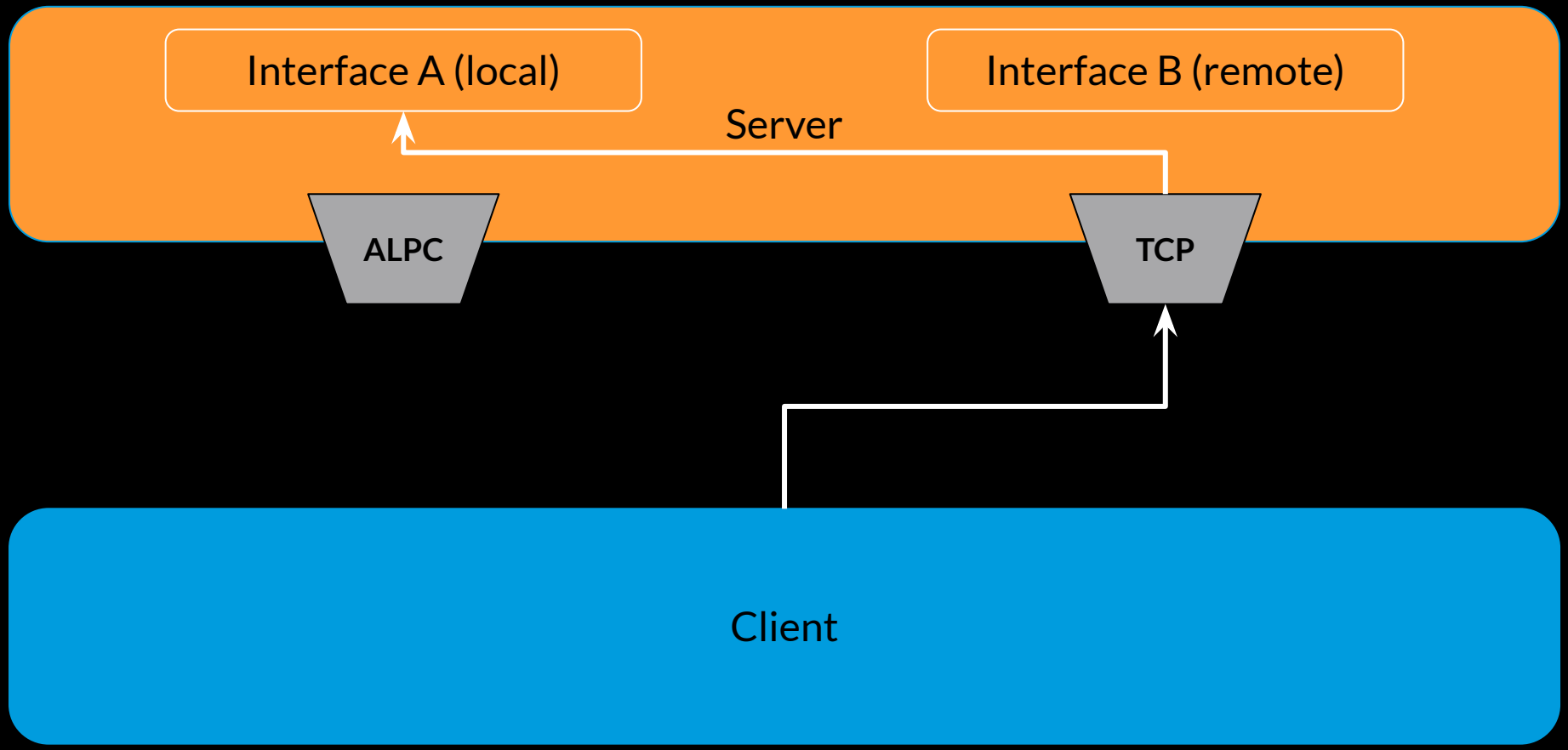
Akamai

“Endpoint Multiplexing”



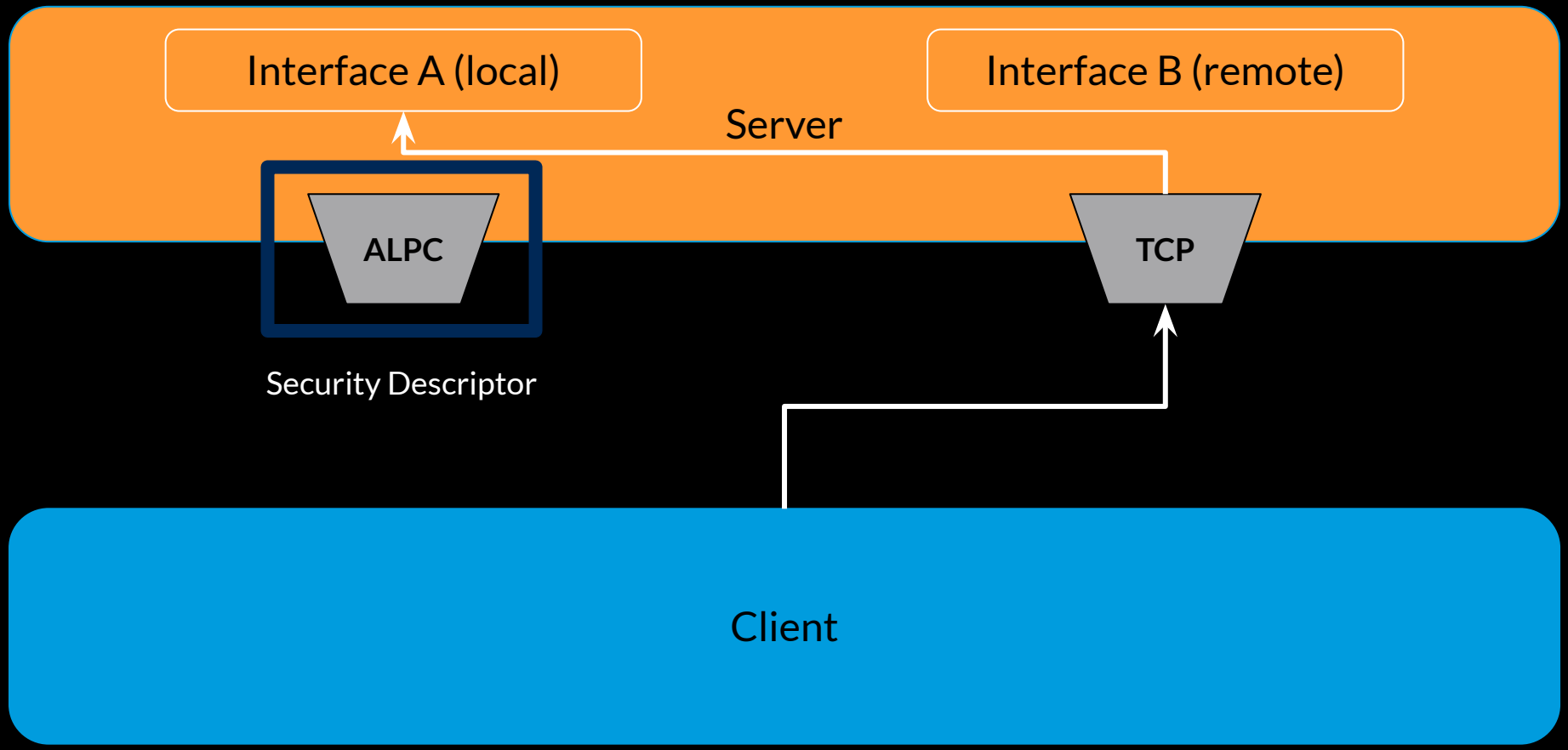
Akamai

“Endpoint Multiplexing”



Akamai

“Endpoint Multiplexing”



Akamai

“Endpoint Multiplexing”

Why?

- Interfaces are not bound to endpoints!

“Endpoint Multiplexing”

Why?

- Interfaces are not bound to endpoints!

When?

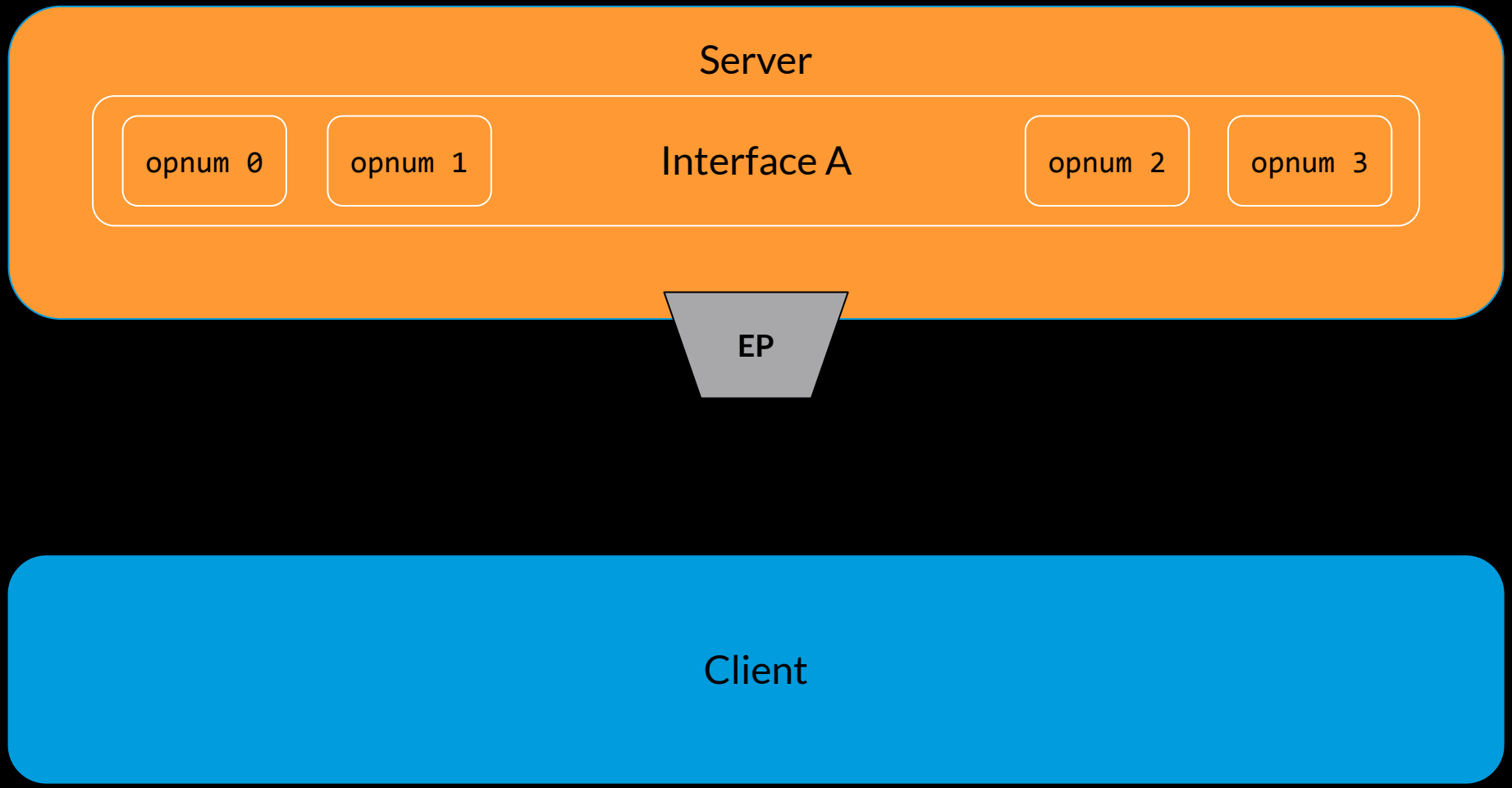
- Service is hosted with other services in the same svchost process

Relevant Flags

🏳 `RPC_IF_ALLOW_LOCAL_ONLY`

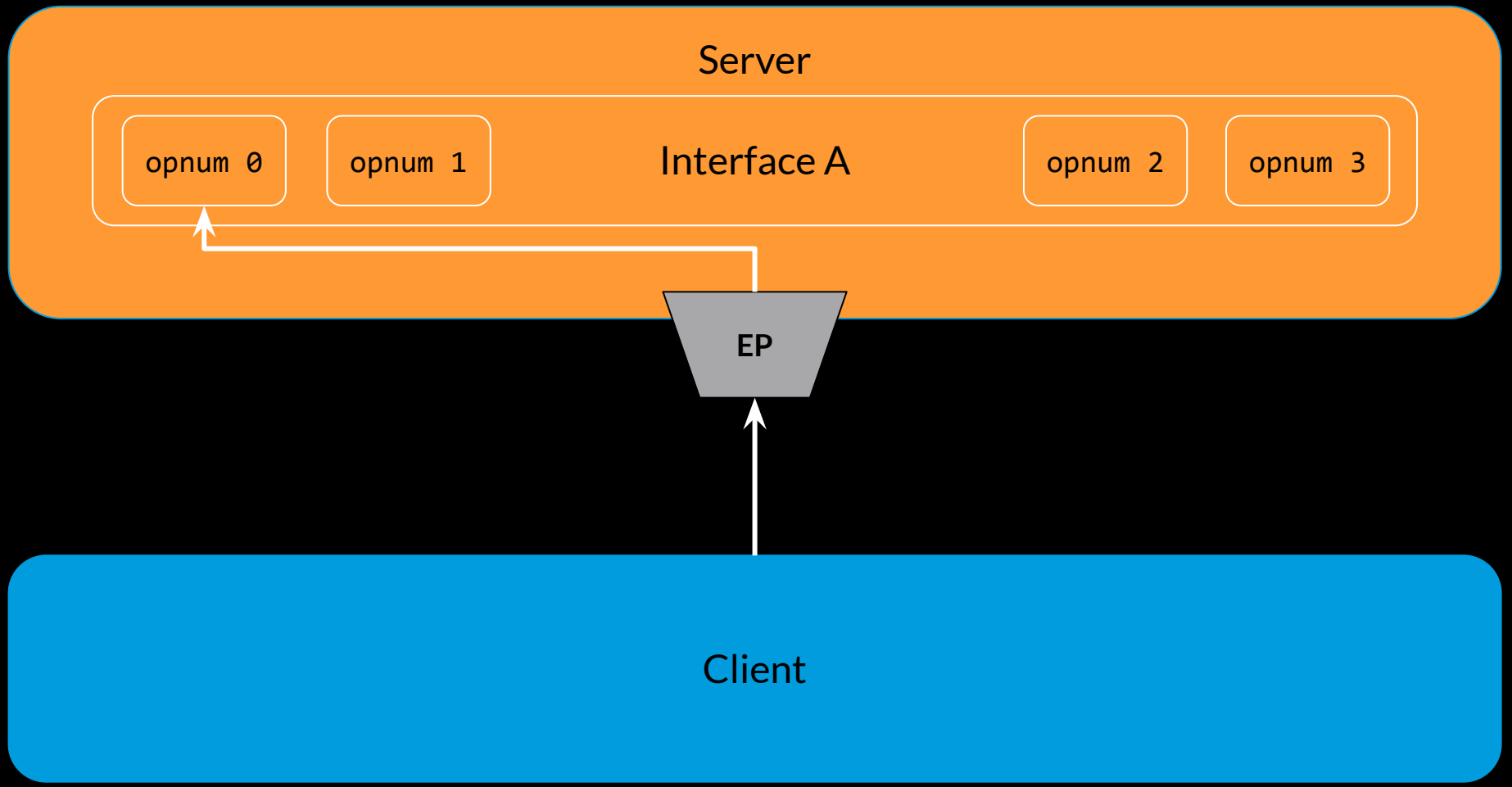
Security Callback Caching

Security Callback Caching



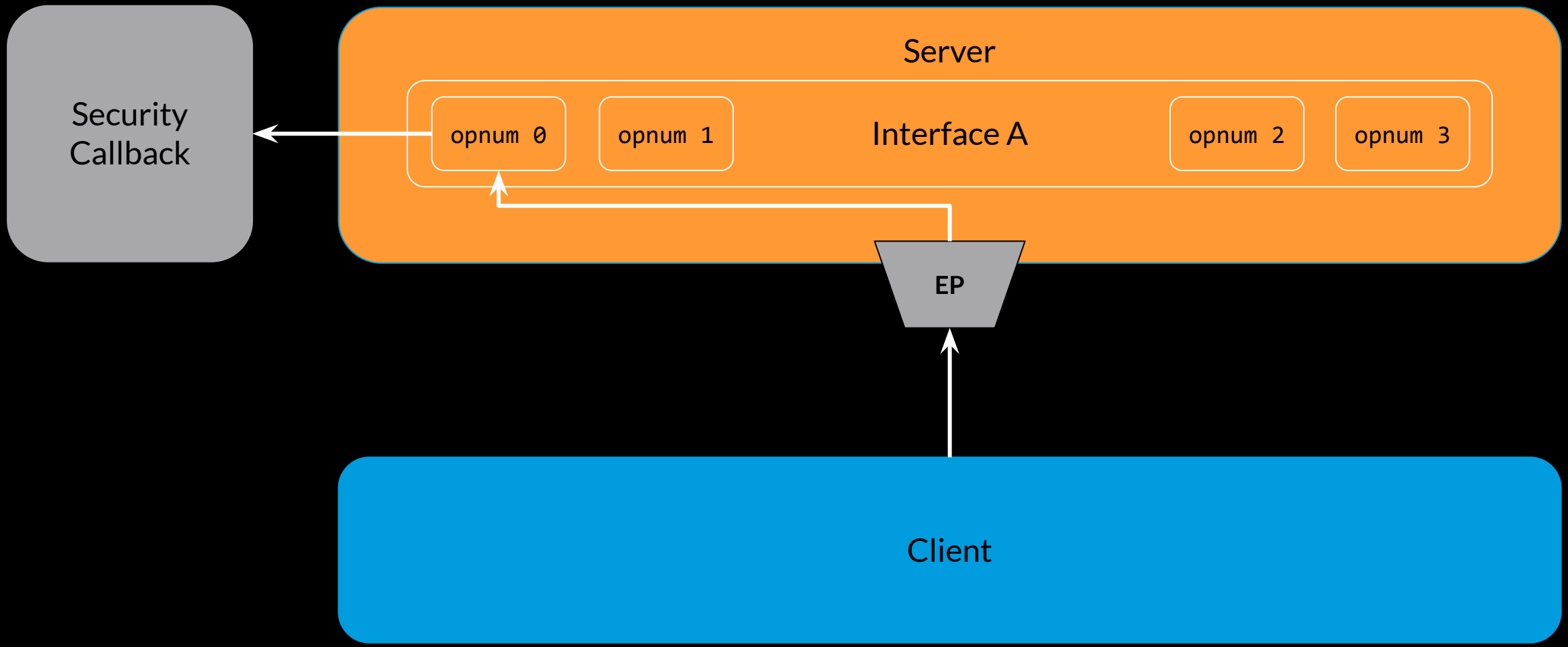
Akamai

Security Callback Caching



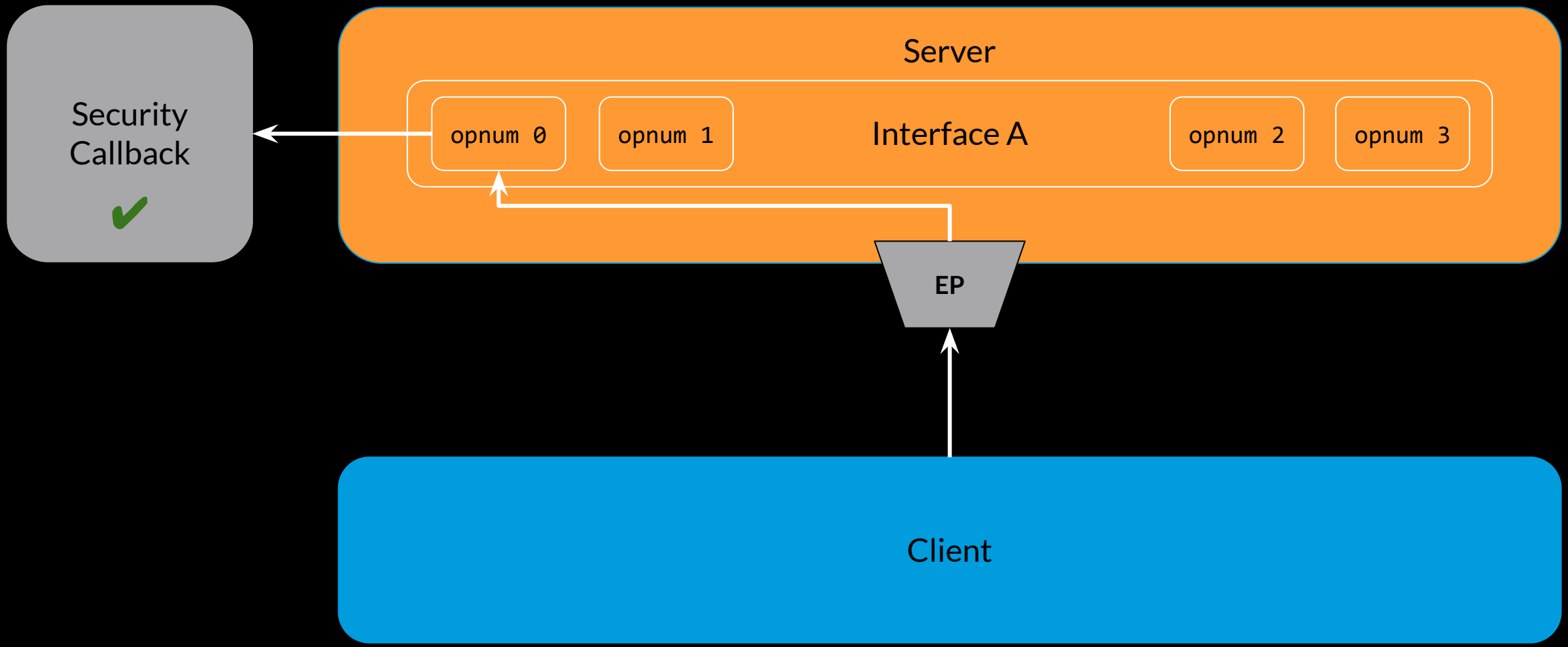
Akamai

Security Callback Caching



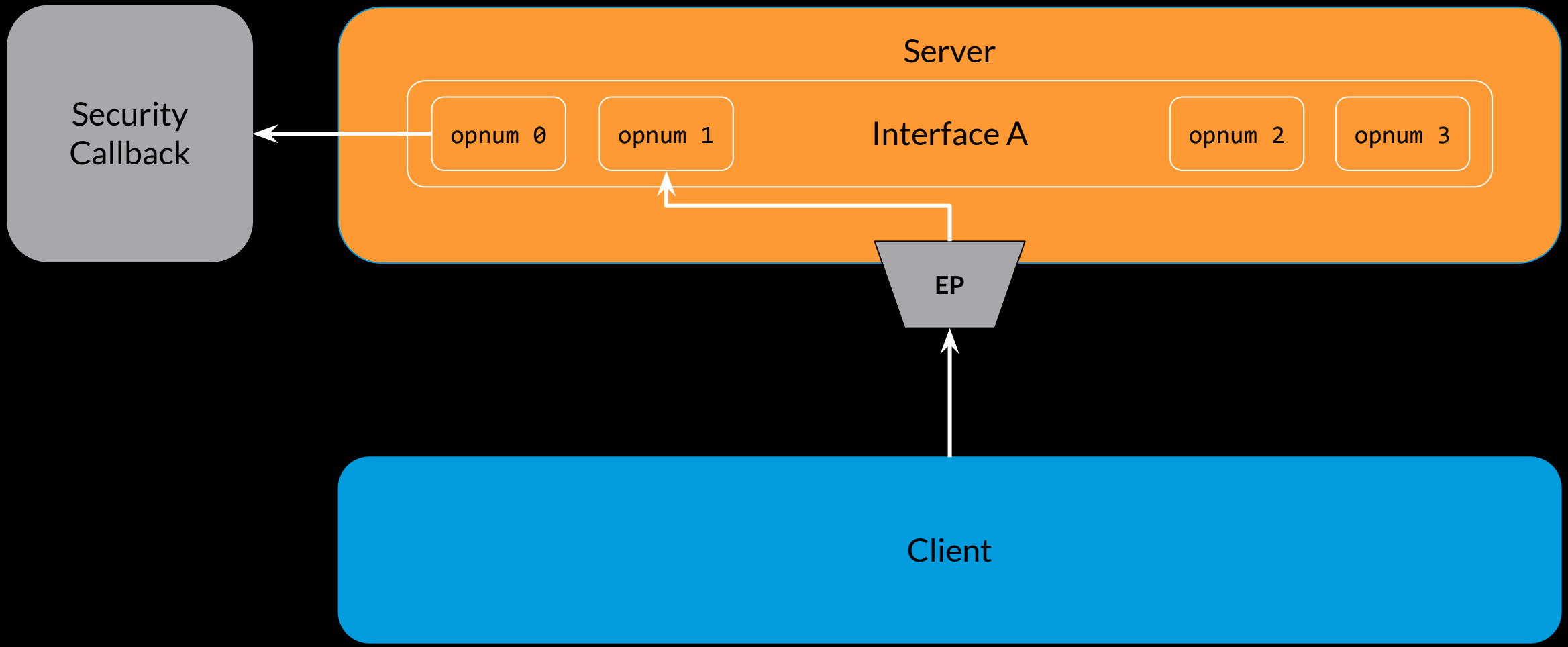
Akamai

Security Callback Caching



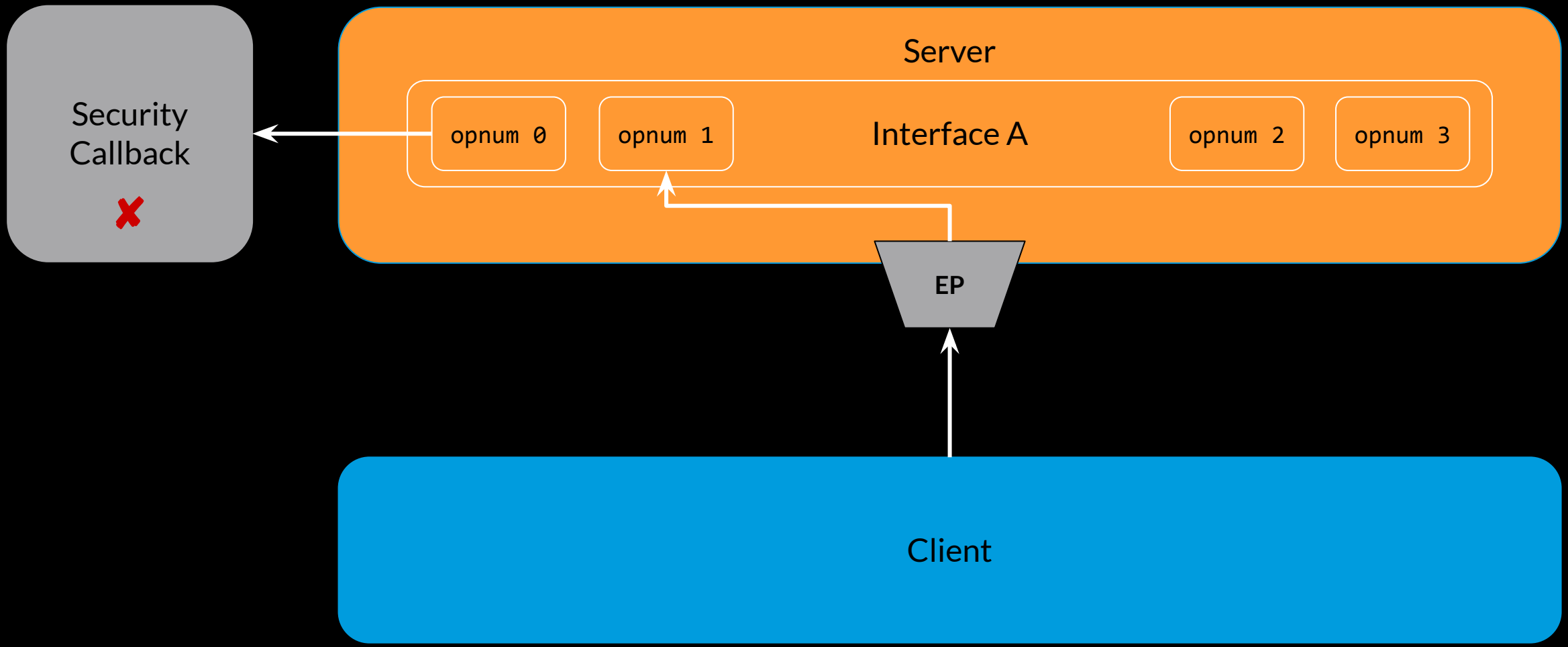
Akamai

Security Callback Caching



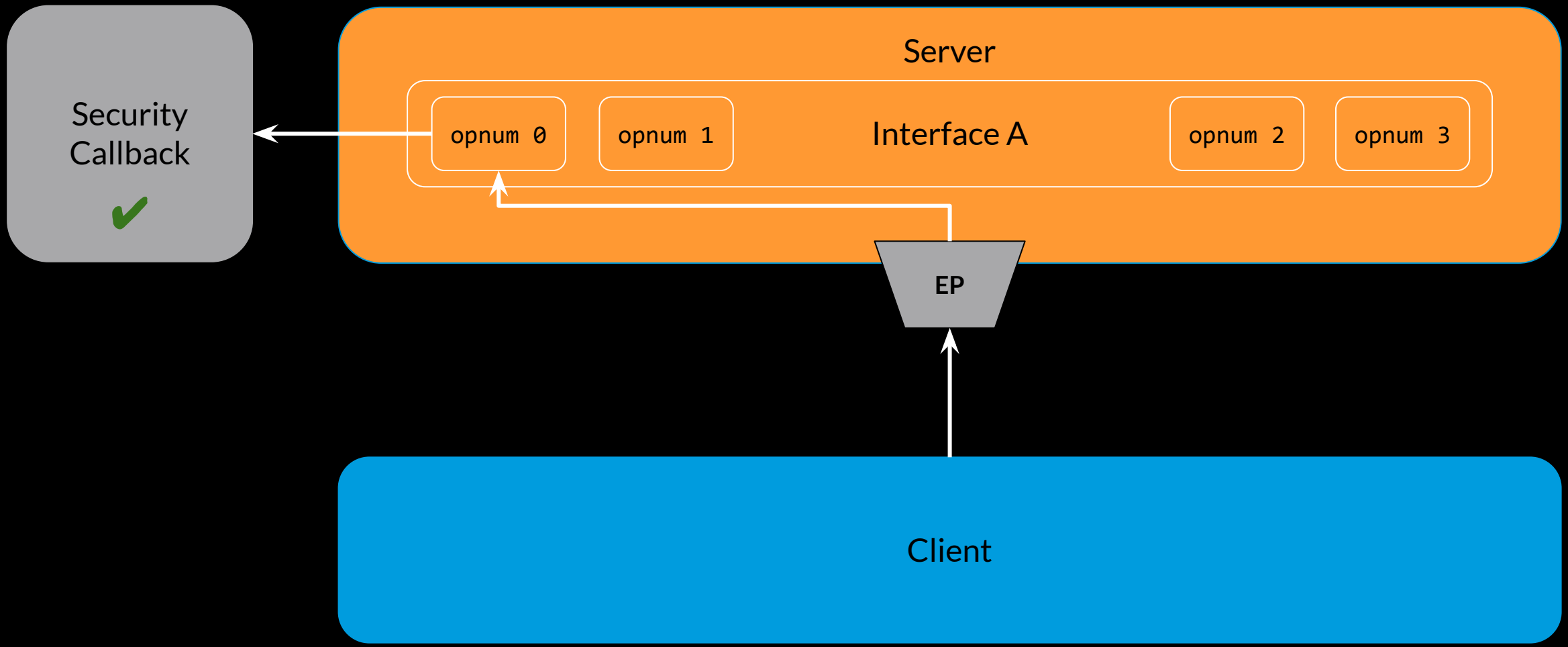
Akamai

Security Callback Caching



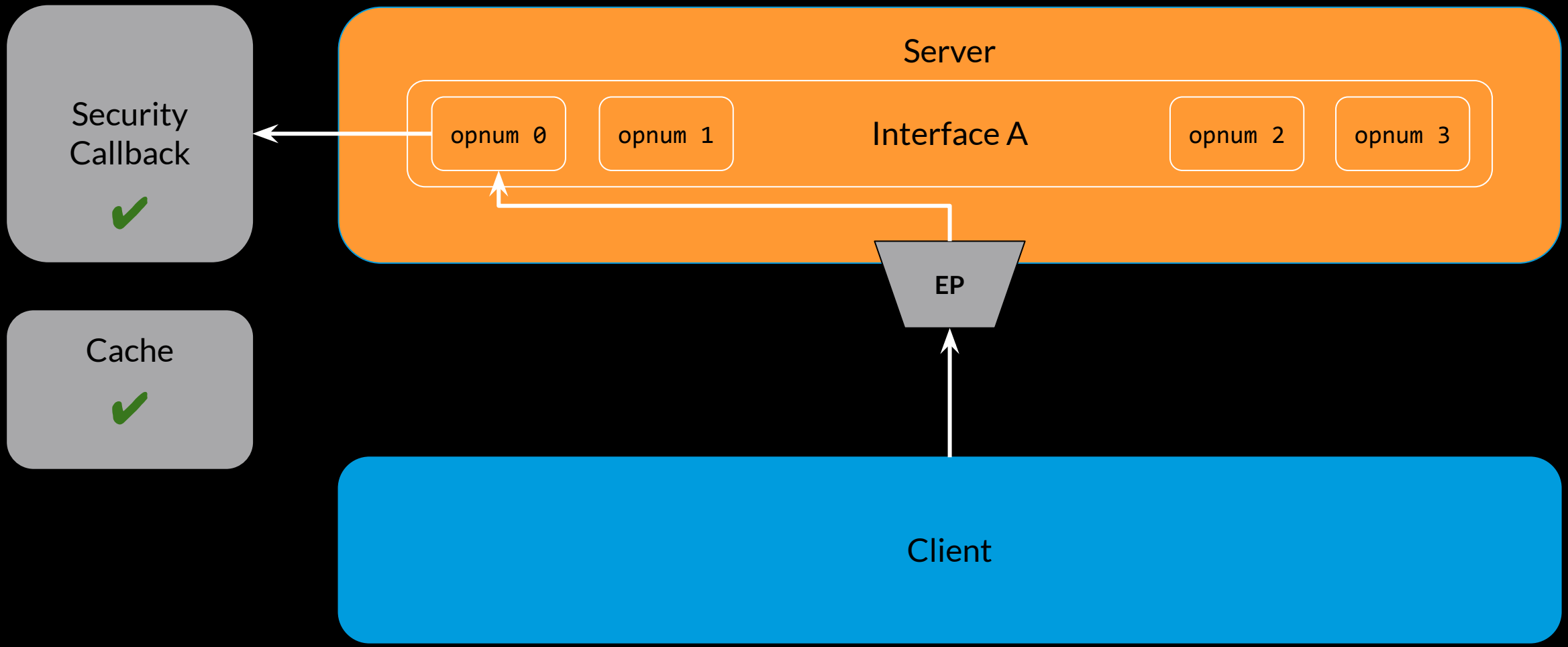
Akamai

Security Callback Caching



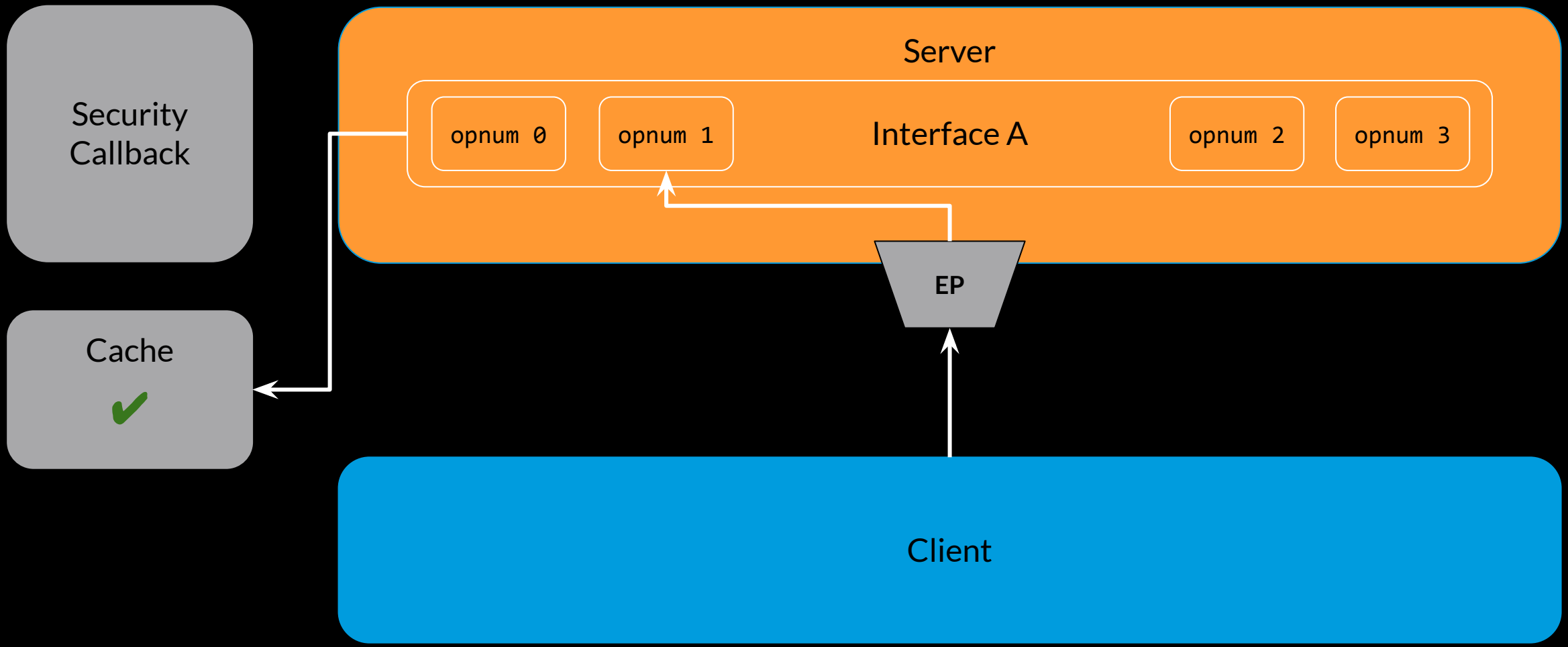
Akamai

Security Callback Caching



Akamai

Security Callback Caching



Akamai

Security Callback Caching

When?

- Happens by default
- relies on the context identifier of the security context

→ Binding not authenticated? no caching!

Relevant Flags

- 🏳 `RPC_IF_SEC_NO_CACHE`
- 🏳 `RPC_IF_SEC_CACHE_PER_PROC`

Quick Recap

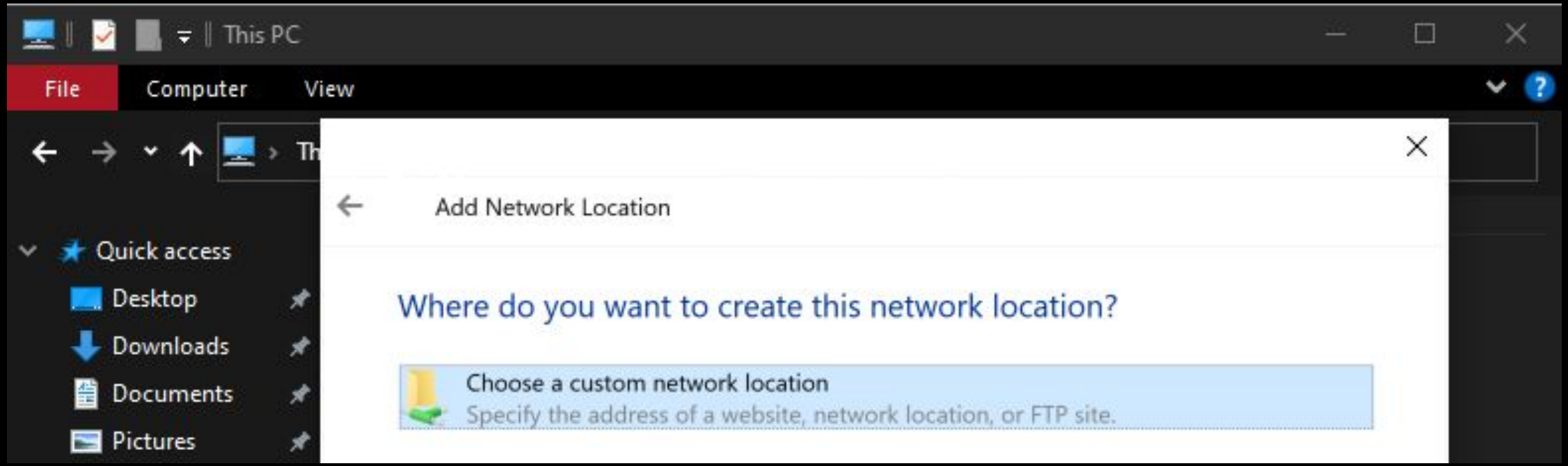
- Authentication Bindings
- Security descriptors
- Security callbacks
- Endpoint “multiplexing”
- Security callback response caching

A 0-Day in the Server service

Bug, attack flow & demo

The Server Service (i.e. *LanmanServer*)

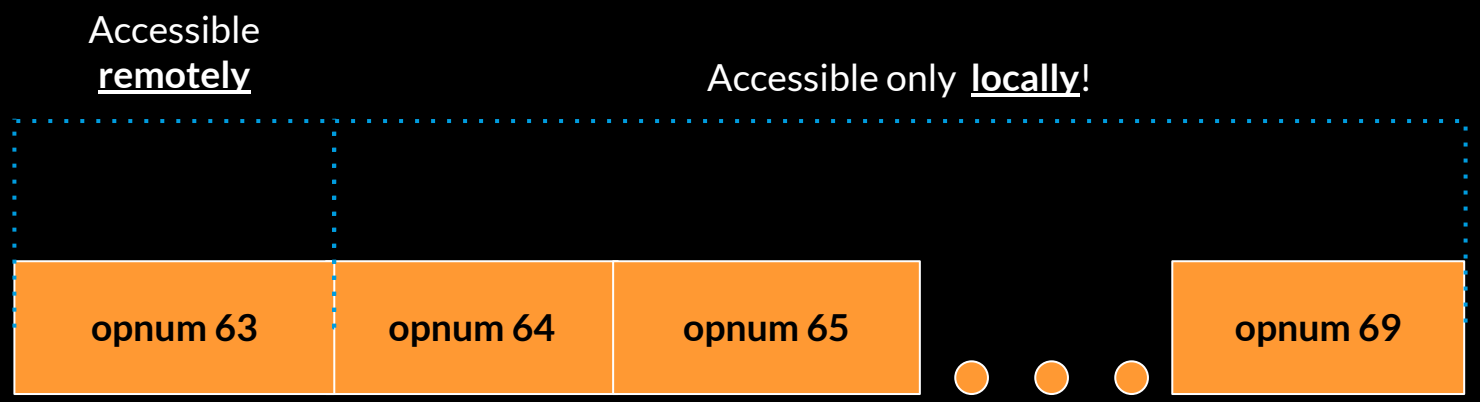
- Accessible through the `\pipe\srvsvc` named pipe



Akamai

Server's Security Callback

```
# Windows 10 19H2
if ((RpcCallAttributes.OpNum - 64) <= 5 && RpcCallAttributes.IsClientLocal != 1))
    return ERROR_ACCESS_DENIED;
```

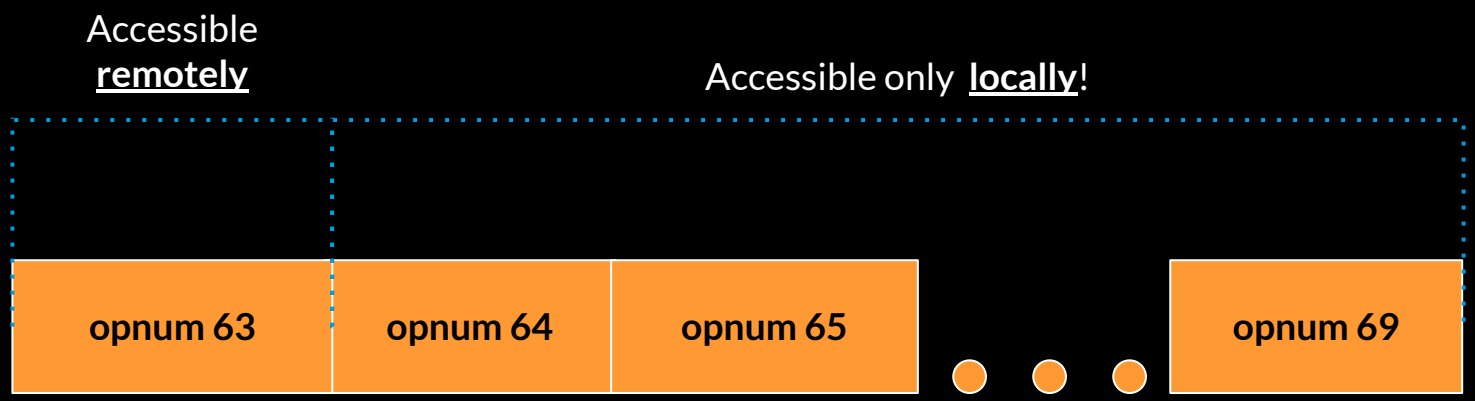


Akamai

Server's Security Callback

Theory - newly added functions can pose a security problem in the future

```
# Windows 10 19H2
if ((RpcCallAttributes.OpNum - 64) <= 5 && RpcCallAttributes.IsClientLocal != 1))
    return ERROR_ACCESS_DENIED;
```

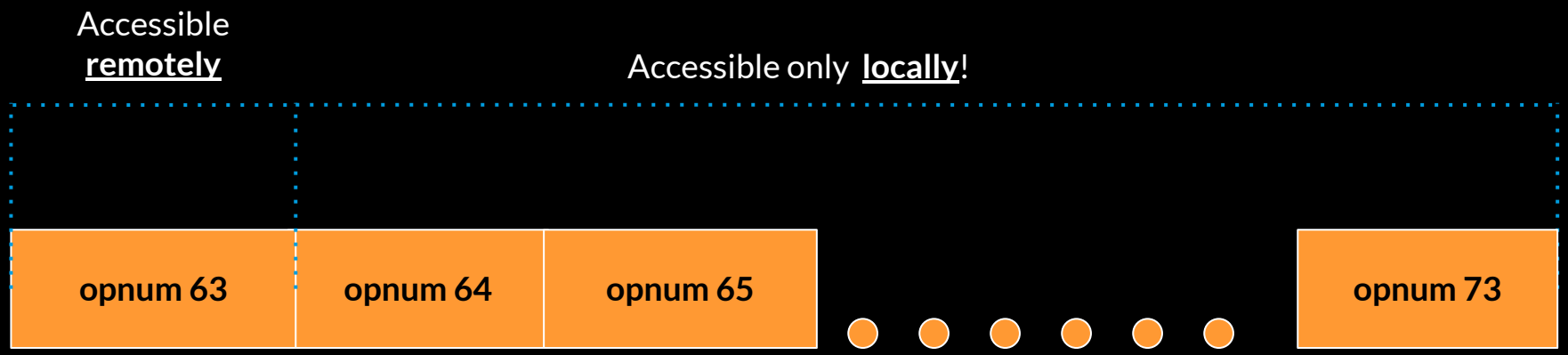


Akamai

Server's Security Callback

Theory - newly added functions can pose a security problem in the future

```
# Windows 10 20H2
if ((RpcCallAttributes.OpNum - 64) <= 9 && RpcCallAttributes.IsClientLocal != 1))
    return ERROR_ACCESS_DENIED;
```

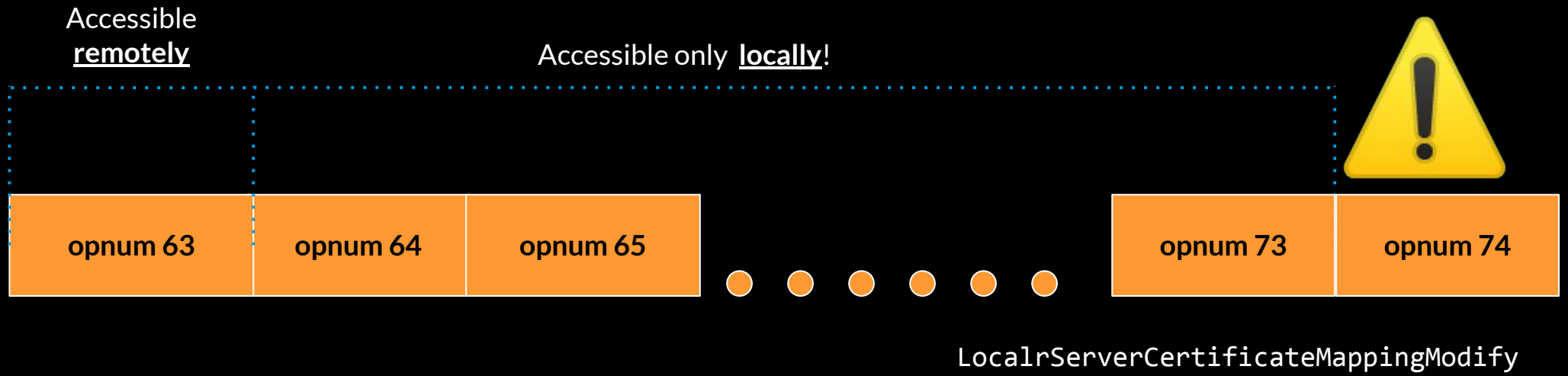


Akamai

Server's Security Callback

Theory - newly added functions can pose a security problem in the future

```
# Windows 11
if ((RpcCallAttributes.OpNum - 64) <= 9 && RpcCallAttributes.IsClientLocal != 1))
    return ERROR_ACCESS_DENIED;
```



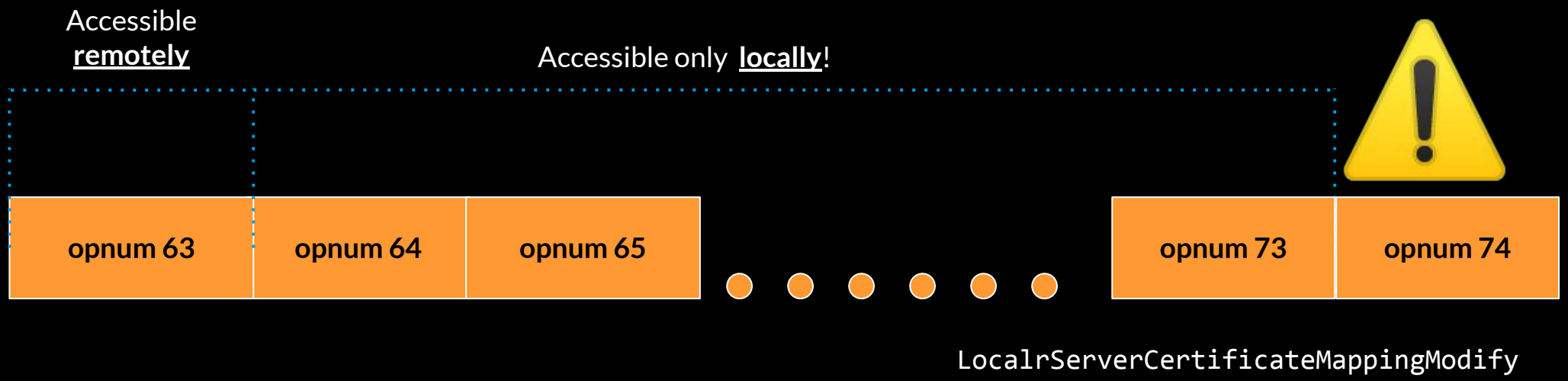
Akamai

Server's Security Callback

CVE-2022-30216 - Tampering (CVSS: 8.8)

Theory - newly added functions can pose a security problem in the future

```
# Windows 11
if ((RpcCallAttributes.OpNum - 64) <= 9 && RpcCallAttributes.IsClientLocal != 1))
    return ERROR_ACCESS_DENIED;
```



Akamai

SMB over QUIC

SMB over QUIC

- Transport layer protocol with low latency, privacy and security

SMB over QUIC

- Transport layer protocol with low latency, privacy and security
- Server provides a certificate - prevents server spoofing attacks

SMB over QUIC

- Transport layer protocol with low latency, privacy and security
- Server provides a certificate - prevents server spoofing attacks
- New functions added - manage the “symbolic link” of a QUIC certificate to a certificate in the certificate store
  - `LocalrServerCertificateMappingGet`
  - `LocalrServerCertificateMappingSet`
  - `LocalrServerCertificateMappingEnum`
  - `LocalrServerCertificateMappingRemove`
  - `LocalrServerCertificateMappingModify`

CVE-2022-30216

- Tampering - we can change a certificate mapping
- Maybe we can do more?

CVE-2022-30216

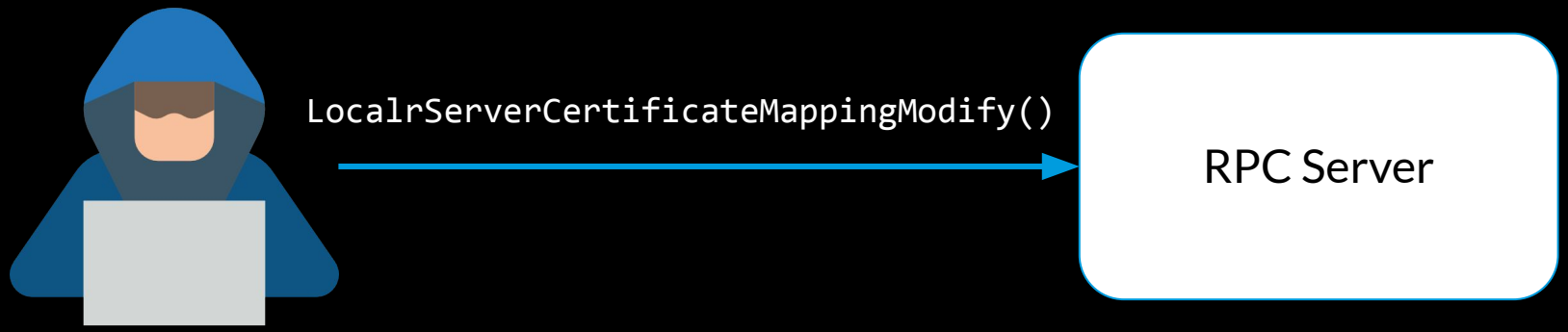
- Tampering - we can change a certificate mapping
- Maybe we can do more?

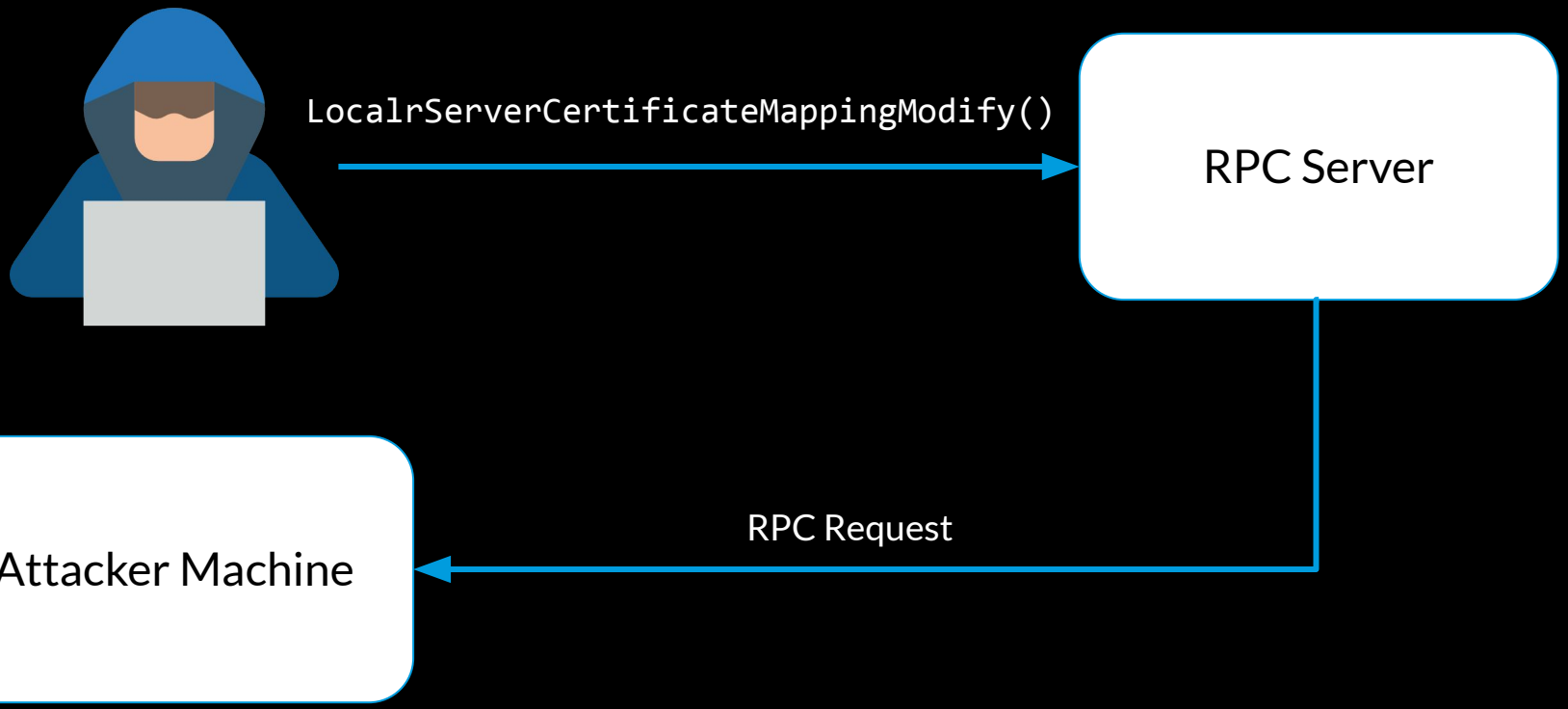
```
00000000
00000000 certificateStruct struc ; (sizeof=0x58, mappedto_93)
00000000 serverName      dq ?                    ; offset
00000008 subject         dq ?                    ; offset
00000010 issuer          dq ?                    ; offset
00000018 thumbprint      dq ?                    ; offset
00000020 friendlyName    dq ?                    ; offset
00000028 notBefore       dq ?                    ; offset
00000030 notAfter        dq ?                    ; offset
00000038 storeLocation   dq ?                    ; offset
00000040 storeName       dq ?                    ; offset
00000048 field_48        dq ?                    ; offset
00000050 type            dd ?
00000054 flags           dd ?
00000058 certificateStruct ends
```

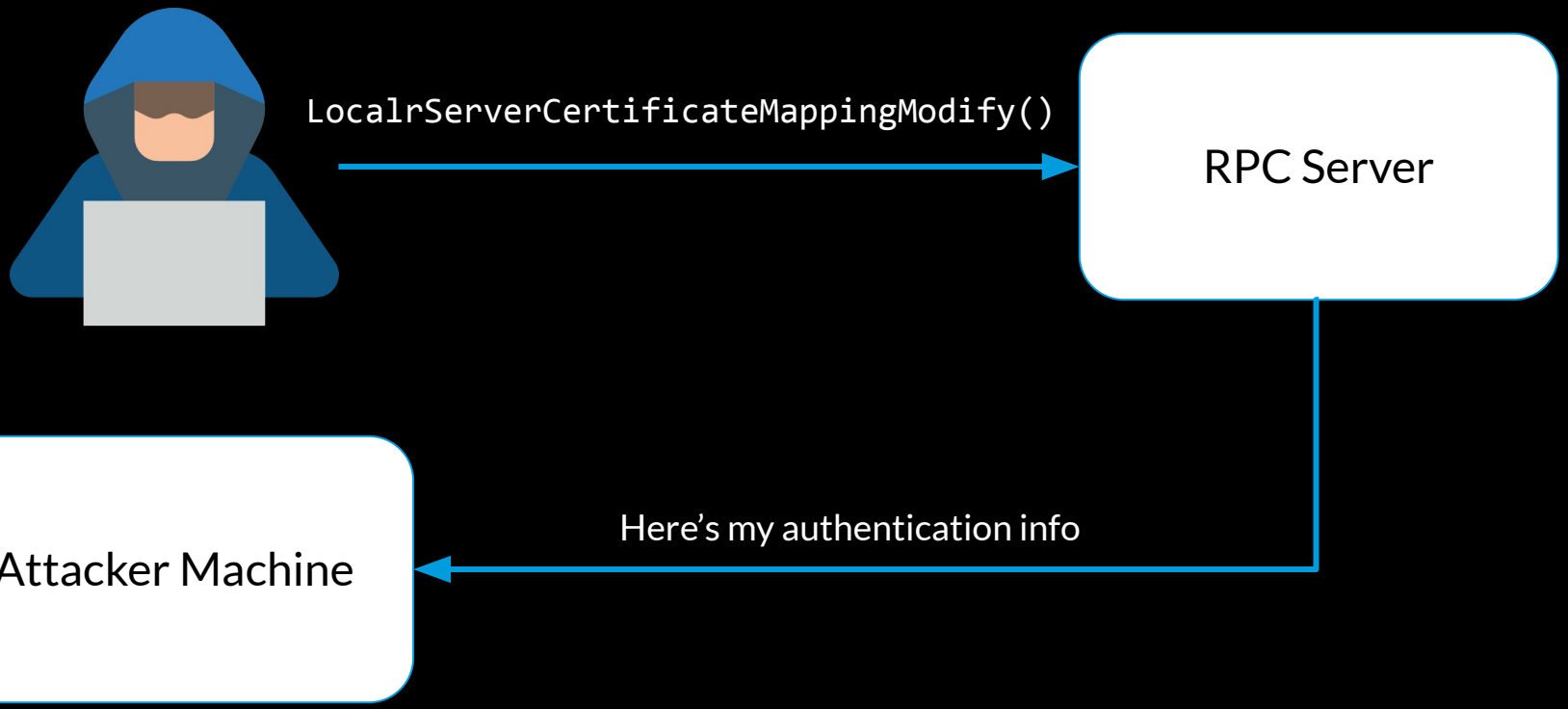
CVE-2022-30216

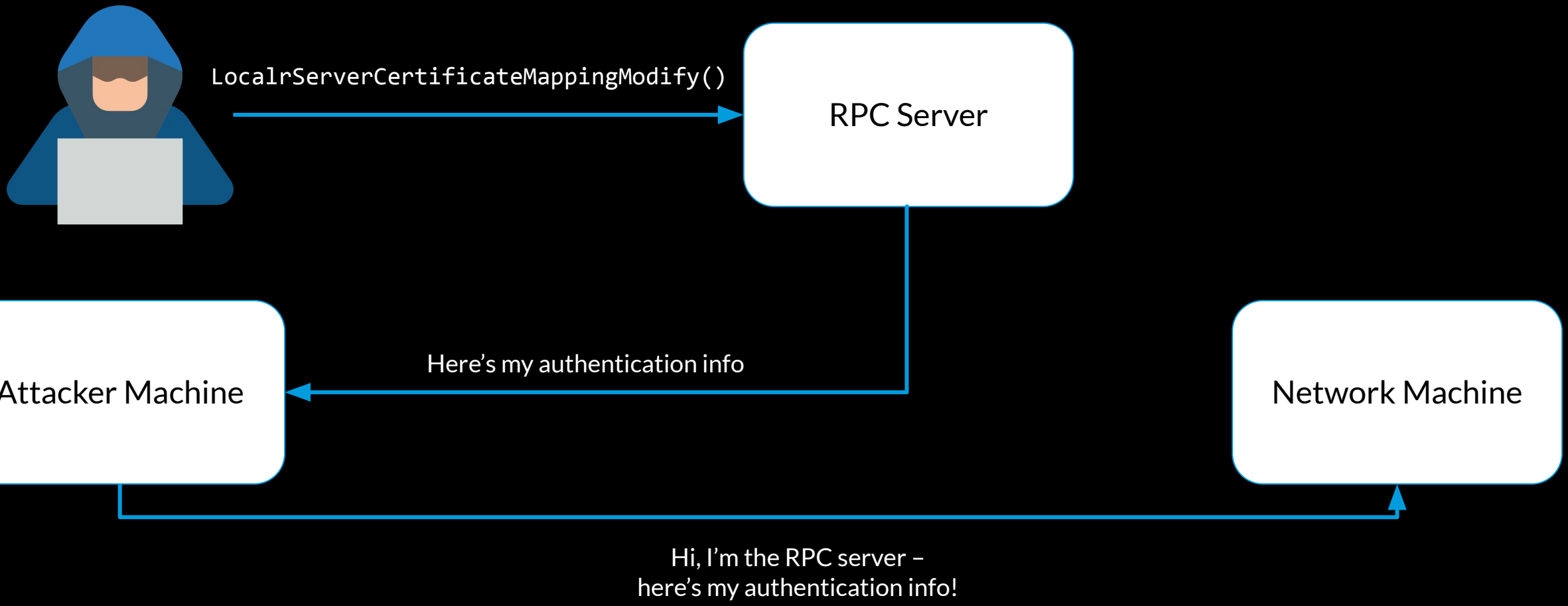
- Tampering - we can change a certificate mapping
- Maybe we can do more?

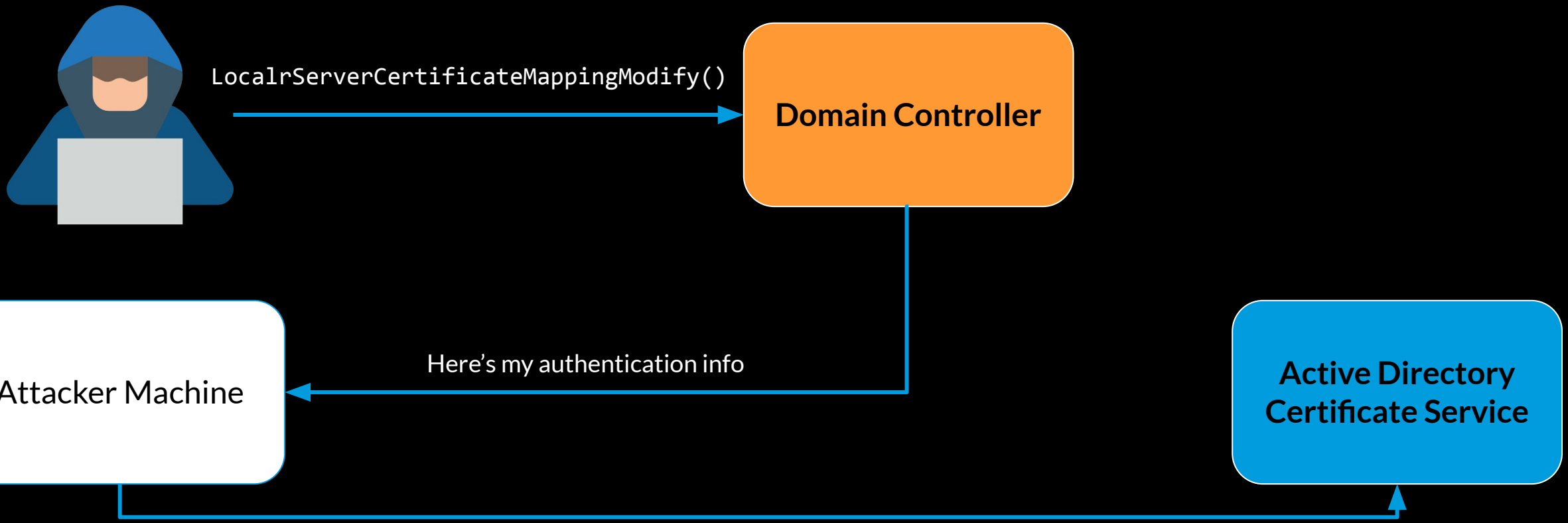
```
00000000
00000000 certificateStruct struc ; (sizeof=0x58, mappedto_93)
00000000 serverName      dq ?                    ; offset
00000008 subject         dq ?                    ; offset
00000010 issuer          dq ?                    ; offset
00000018 thumbprint      dq ?                    ; offset
00000020 friendlyName    dq ?                    ; offset
00000028 notBefore       dq ?                    ; offset
00000030 notAfter        dq ?                    ; offset
00000038 storeLocation   dq ?                    ; offset
00000040 storeName       dq ?                    ; offset
00000048 field_48        dq ?                    ; offset
00000050 type            dd ?
00000054 flags           dd ?
00000058 certificateStruct ends
```



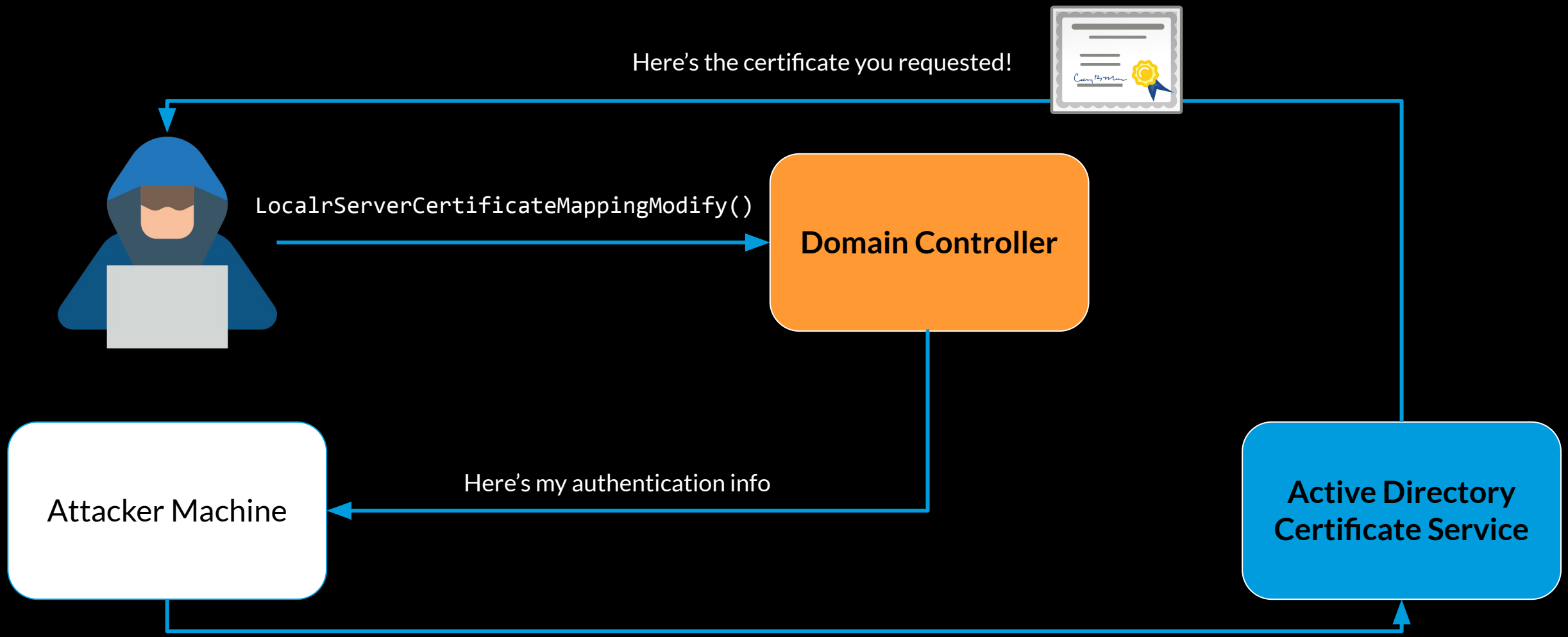

LocalrServerCertificateMappingModify()
RPC Server
RPC Request
Attacker Machine


LocalrServerCertificateMappingModify()
RPC Server
Here's my authentication info
Attacker Machine


LocalrServerCertificateMappingModify()
RPC Server
Attacker Machine
Here's my authentication info
Network Machine
Hi, I'm the RPC server –
here's my authentication info!



Hi, I'm the **Domain Controller** –
here's my authentication info!
© Will Schroeder & Lee Christensen


Here's the certificate you requested!
LocalrServerCertificateMappingModify()
Domain Controller
Attacker Machine
Here's my authentication info
Active Directory Certificate Service
Hi, I'm the Domain Controller –
here's my authentication info!
© Will Schroeder & Lee Christensen

Akamai

Exploitation Flow

Action	Result
Auth coerce by calling vulnerable RPC function	Victims sends out credentials

Exploitation Flow

Action	Result
Auth coerce by calling vulnerable RPC function	Victims sends out credentials
Relay credentials to ADCS	ADCS outputs a certificate

Exploitation Flow

Action	Result
Auth coerce by calling vulnerable RPC function	Victims sends out credentials
Relay credentials to ADCS	ADCS outputs a certificate
Use Rubeus with certificate	Computer Kerberos TGT is granted

Exploitation Flow

Action	Result
Auth coerce by calling vulnerable RPC function	Victims sends out credentials
Relay credentials to ADCS	ADCS outputs a certificate
Use Rubeus with certificate	Computer Kerberos TGT is granted
Perform DCSync	NTLM hash is obtained

Exploitation Flow

Action	Result
Auth coerce by calling vulnerable RPC function	Victims sends out credentials
Relay credentials to ADCS	ADCS outputs a certificate
Use Rubeus with certificate	Computer Kerberos TGT is granted
Perform DCSync	NTLM hash is obtained
Pass the hash	**Get shell**

Exploit Demo

```
C:\Users\Administrator>ipconfig

Windows IP Configuration


Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::6085:c673:f1e2:de25%9
   IPv4 Address. . . . . . . . . . . : 10.0.200.132
   Subnet Mask . . . . . . . . . . . : 255.255.0.0
   Default Gateway . . . . . . . . . : 10.0.0.138

C:\Users\Administrator>whoami
research\administrator

C:\Users\Administrator>_
```

Summary

- Security callbacks are an interesting attack surface
  - Specifically dealing with opnums
  - Specifically due to caching
- Future research directions
  - More services, SMB over QUIC, RPC runtime, tooling
- Blog post & PoC available at https://akamai.com/blog/security/

References

- Offensive Windows IPC Internals 2: RPC (0xcsandker)
- How to secure a Windows RPC Server, and how not to (@tiraniddo)
- ADCS + PetitPotam NTLM Relay: Obtaining krbtgt Hash with Domain Controller Machine Certificate

Thank you

Questions?

@nachoskrnl @OphirHarpaz