

MS RPC Architecture & Security Problems Related

kkqq (Lin Yichong)

(kk_qq@263.net kkqq@SST, kkqq@USTC)

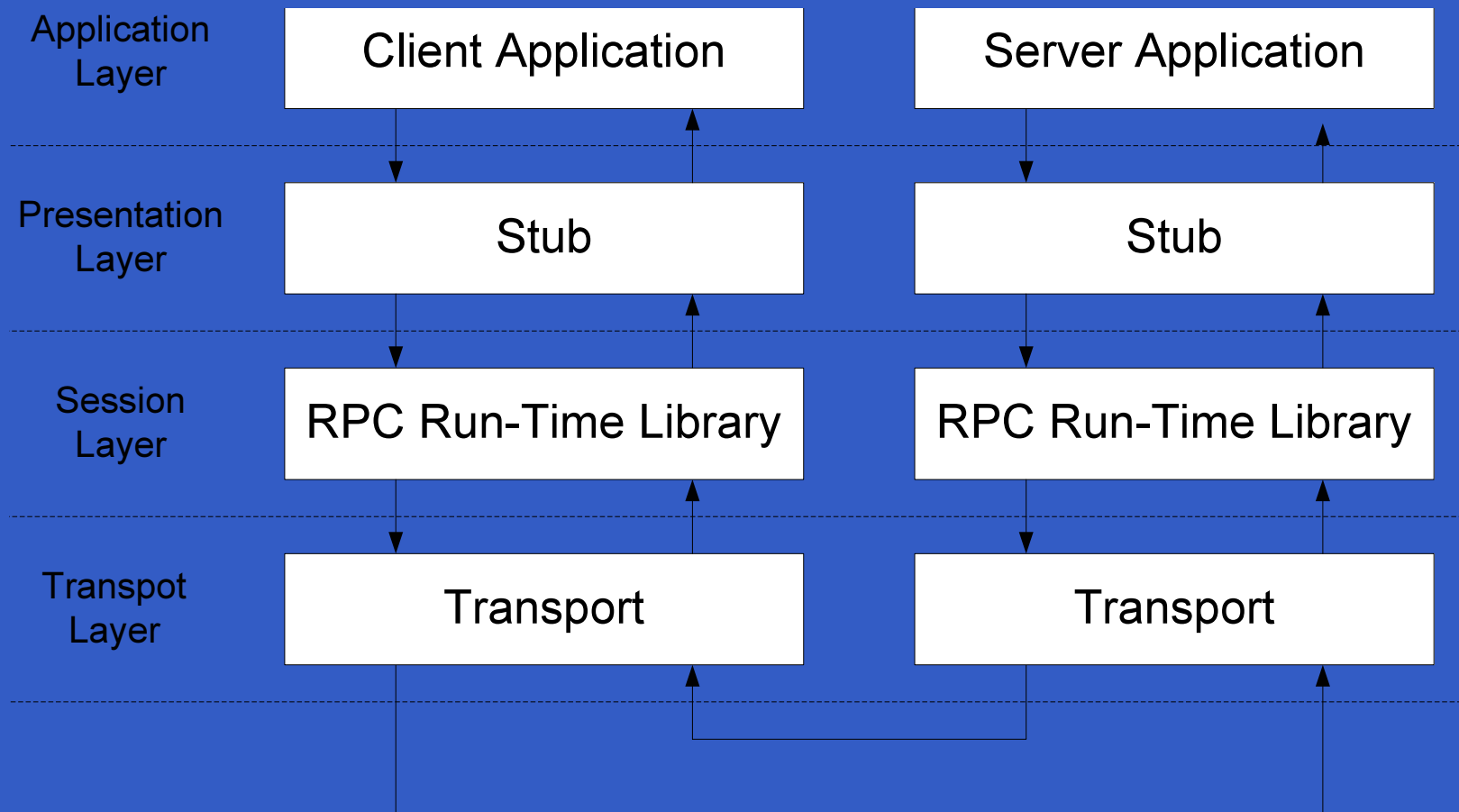
<http://www.0x557.org> <http://kkqq.blogdns.com>

Powered by \LaTeX and prosper.

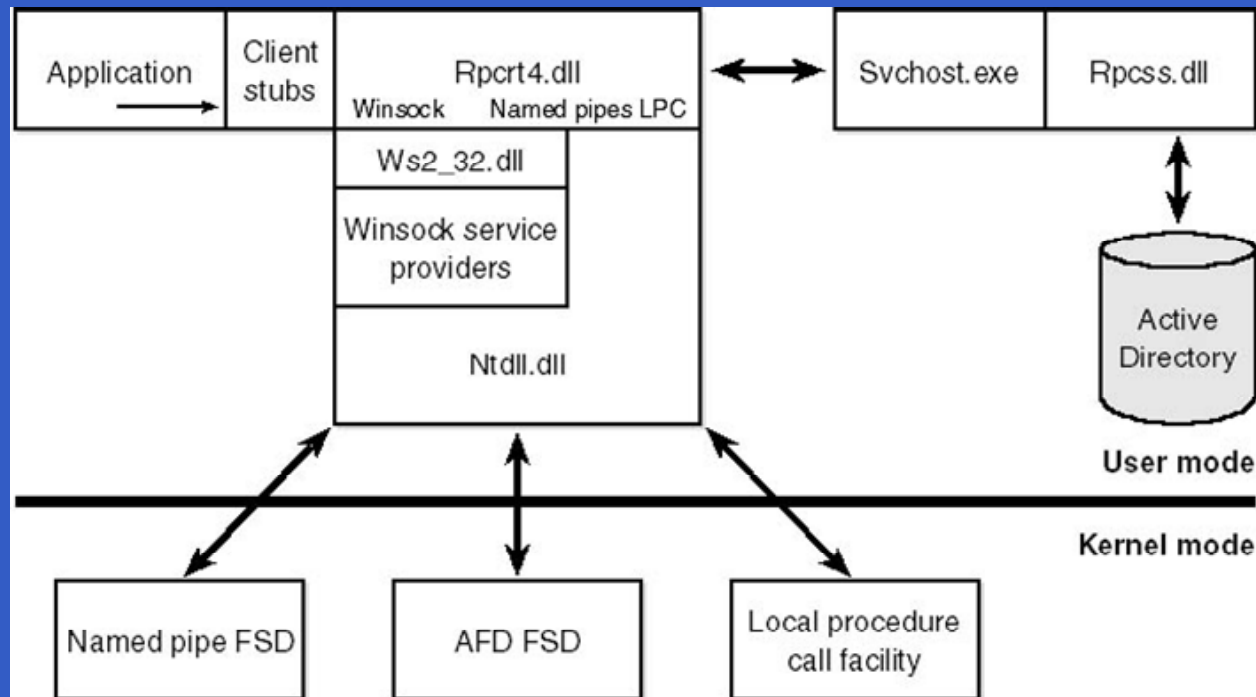
Content

- MS RPC Architecture & Data Flow
- Hacking the Binary for Fun and Profit :)
- Security Problems in MS RPC

RPC – Remote Procedure Call



RPC – Implement



From Inside win2k 3rd Edition

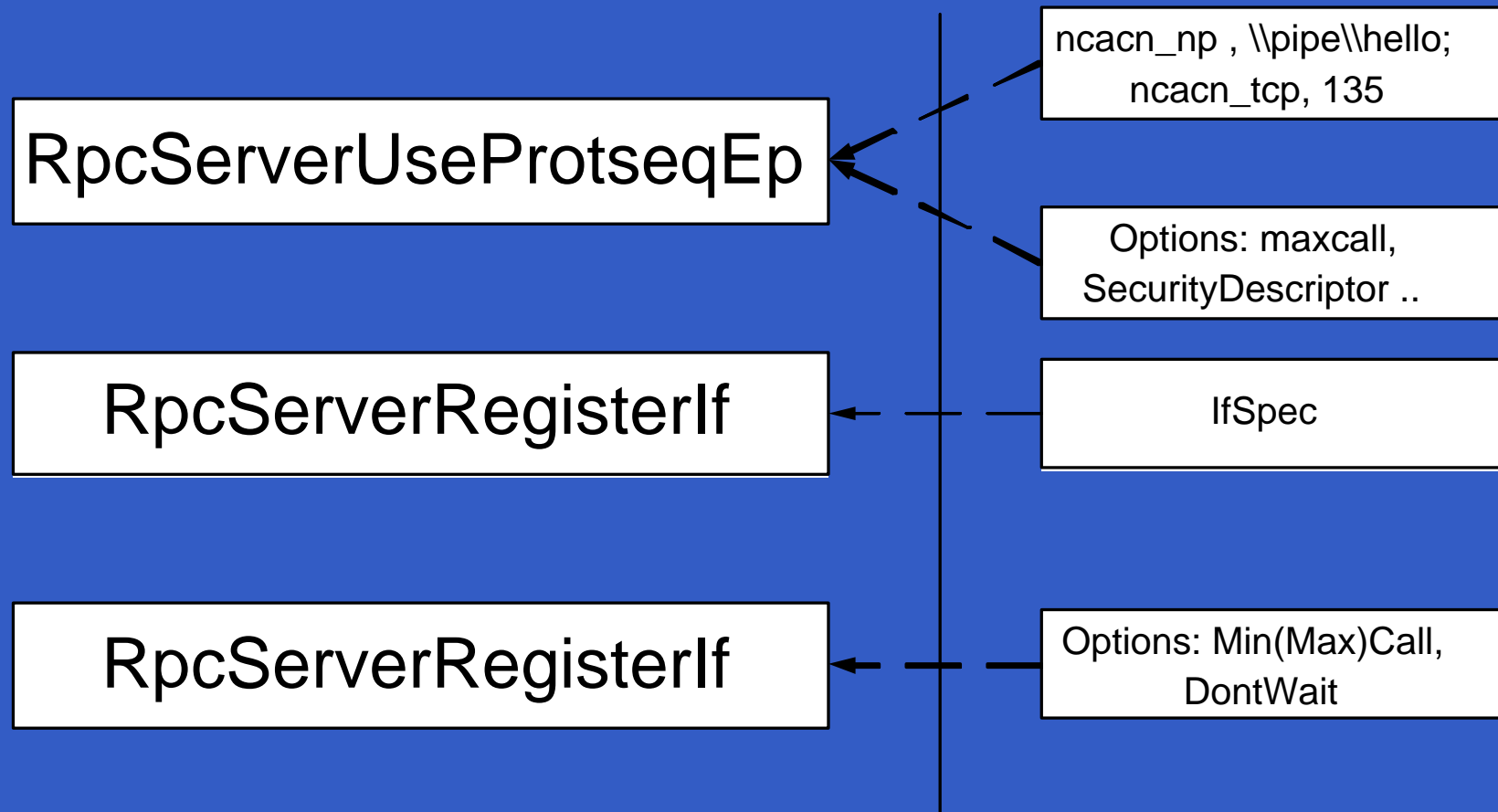
Services:

- Endpoint Mapper (135)
- Remote Management Interface (1025)

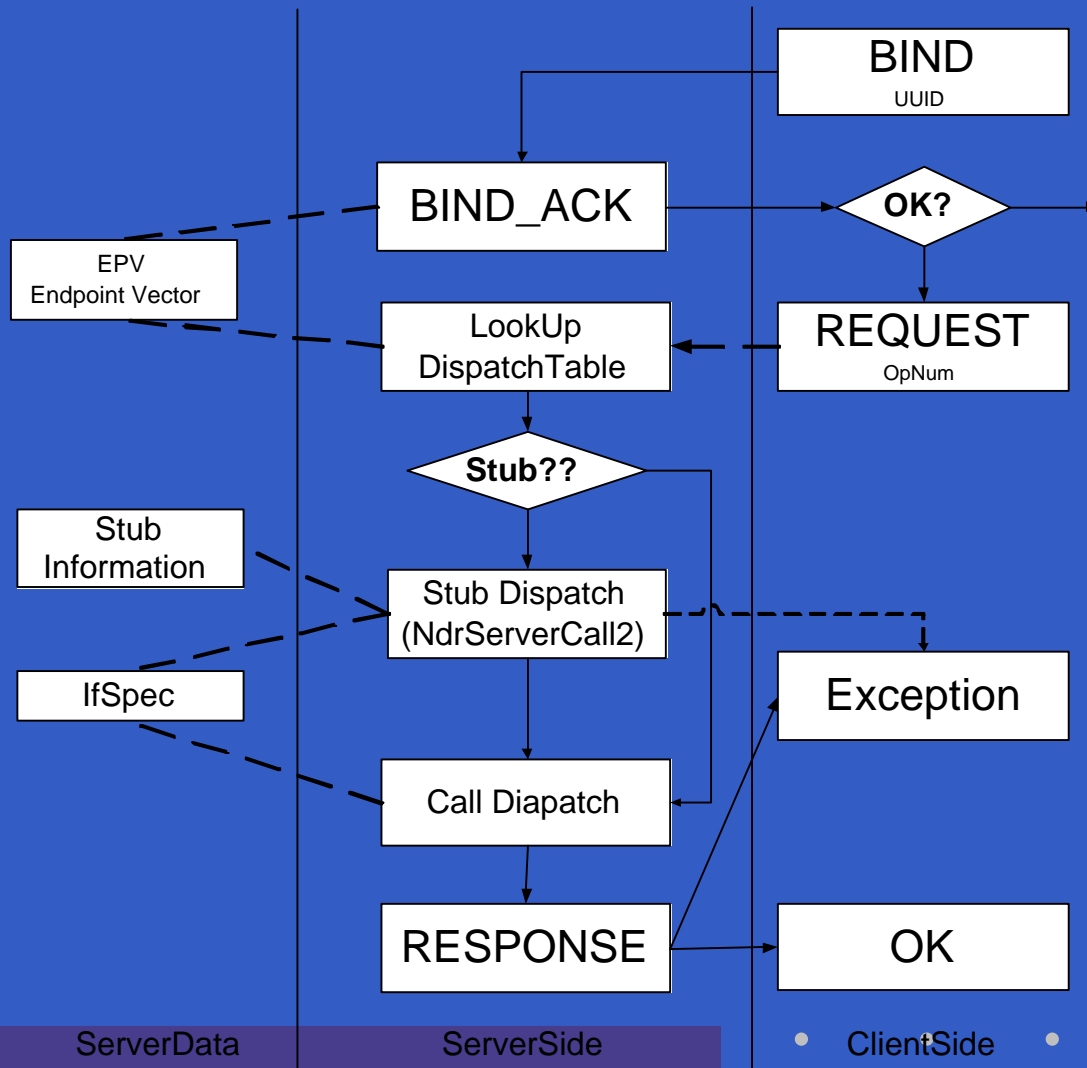
RPC – Programming

- Define the interface(.idl files)
- Generate Stub Code(midl xxx.idl)
- Implement the Interface
- Note: Endpoint
 - TCP
 - UDP
 - SMB
 - Named Pipe
 - HTTP ...

Data Flow – Server Listening

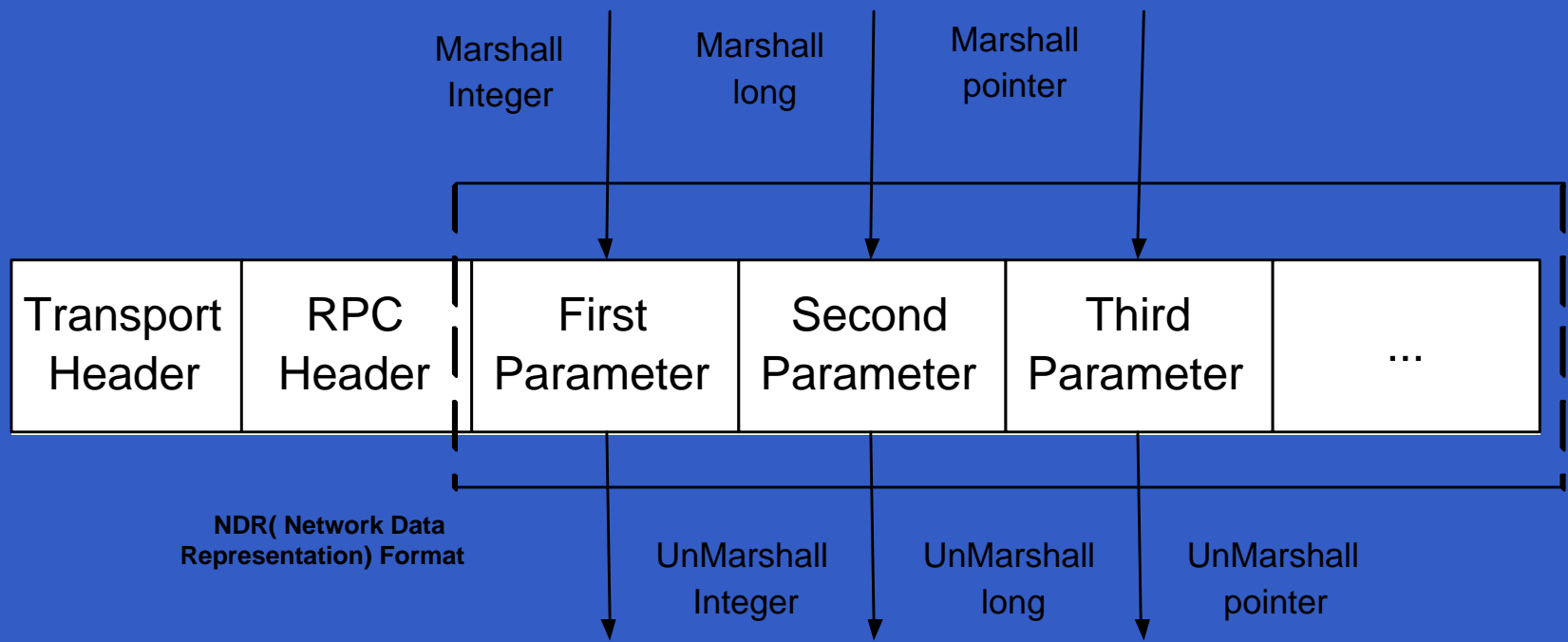


Data Flow – Call the Dispatch Routine



Data Flow – Marshall and UnMarshall

RpcClientCall(uint_32 first, ulong second, pointer third, . . .)



RpcServerCall(uint_32 first, ulong second, pointer third, . . .)

Hacking the Binary for Fun and Profit

Purpose: How the application dispatch different RPC request.

Current Problems:

- Server Application may be runned over different Endpoint.
- Find the real data flow with debugging dynamicly.

Samples:<http://www.xfocus.net/articles/200311/641.html>

- Complex data-struct & and data-flow.

Our Method: IDC Script file

Data struct of Application Server-side

```
typedef struct _RPC_SERVER_INTERFACE
```

```
{
  unsigned int
  RPC_SYNTAX_IDENTIFIER
  RPC_SYNTAX_IDENTIFIER
  PRPC_DISPATCH_TABLE
  unsigned int
  PRPC_PROTSEQ_ENDPOINT
  RPC_MGR_EPV__RPC_FAR *
  void const __RPC_FAR *
  unsigned int
} RPC_SERVER_INTERFACE;
```

```
Length;
Interfaceld;
TransferSyntax;
DispatchTable;
RpcProtseqEndpointCount;
RpcProtseqEndpoint;
DefaultManagerEpv;
InterpreterInfo;
Flags;
```

```
typedef struct {
  unsigned int
  RPC_DISPATCH_FUNCTION
  int
} RPC_DISPATCH_TABLE
```

```
DispatchTableCount;
DispatchTable;
Reserved;
```

With midl/Oicf

NULL

With midl/Oicf

Dispatch Table
NdrServerCall2
NdrServerCall2
...

```
typedef struct _MIDL_SERVER_INFO_
```

```
{
  PMIDL_STUB_DESC
  const SERVER_ROUTINE *
  PFORMAT_STRING
  const unsigned short *
  const STUB_THUNK *
  PFORMAT_STRING
  PFORMAT_STRING
  const unsigned short *
} MIDL_SERVER_INFO_;
```

```
pStubDesc;
DispatchTable;
ProcString;
FmtStringOffset;
ThunkTable;
LocalFormatTypes;
LocalProcString;
LocalFmtStringOffset;
```

With midl/Oicf

Dispatch Table
Entry 1
Entry 2
...

Samples: rpcss.dll – Endpoint Mapper

```
Find RpcServerRegisterIf at addr at 0x76152012
IfSpec at address 0x7615d400
Interface UUID:    e1af8308-5d1f-11c9-91a4-08002b14a0fa
Dispatch Table:   0x7615d428
    Dispatch Table Count:    7
    Dispatch Table Addr:    0x7615d408
        Entry    0: NdrServerCall2
            ...
MIDL Server Info at address 0x761344b0
DispatchTable:        0x76134480
    Entry    0    loc_76152A35
    Entry    1    loc_76152F30
    Entry    2    sub_7615317C
    Entry    3    sub_7615320D
    Entry    4    loc_761533C2
    Entry    5    loc_76153301
    Entry    6    loc_7615331F
```

Features

- IDC script for IDA Pro Disassembler.
- Verbose and Simple Output.
- Renaming the Dispatch Routine automatically.

Security Problems

- DoS
- Buffer Overrun
- Information Disclosure

Catalog – Where is the vulnerability

- Server-side Application Over MS RPC
 - Messenger Service
 - IIS(Exchange Server...)...
- Implement of MS RPC Protocol
- MS RPC Services
 - EndPoint Mapper (rpcss.dll)
 - DCOM
 - Locator (locator.exe)

Server-side Application Over MS RPC

- Result: DoS or Priviledge Escalation
- Samples
 - Microsoft Exchange 2000 Multiple MSRPC Denial Of Service Vulnerabilities (2002 bid:5421 SPIKE 2.5)
 - Microsoft Windows Message Queuing Service Heap Overfbw Vulnerability (2003 bid:8783)
 - Microsoft Messenger Service Buffer Overrun Vulnerability (2003 bid:8826 LSD)
 - Microsoft Windows Workstation Service Remote Buffer Overfbw Vulnerability (2003 bid:9011 eeye)

Implement of MS RPC Protocol

- Result: DoS
- Samples
 - NT RPC CPU Utilization Vulnerability (1997 bid:688)
 - Microsoft Windows NT RPC DoS Vulnerability (1998 bid:2234)
 - Microsoft Windows 2000 Malformed RPC Packet DoS Vulnerability (1998 bid:1673)
 - Microsoft Windows RPC Service Denial of Service Vulnerability (2002 bid:6005 SPIKE 2.7)
 - Microsoft Malformed RPC Packet Buffer Overfbw Vulnerability (2002 bid:5879)
 - Microsoft Windows 2000 RPC Service Privilege Escalation Vulnerability (2003 bid:6796 SPIKE 2.8)

MS RPC Services (1)

- Result: DoS or Priviledge Escalation
- Samples
 - Microsoft Windows NT RPC Endpoint Mapper Denial of Service Vulnerability (2001 bid:3313)
 - Microsoft Windows RPC Service Denial of Service Vulnerability (2002 bid:6005 svchost SPIKE 2.7)
 - Microsoft Windows Locator Service Buffer Overflw Vulnerability (2003 bid:6666 ngssoftware :P 6666)
 - Microsoft Windows DCOM RPC Interface Buffer Overrun Vulnerability (2003 bid:8205 LSD Blaster WORM!!)
 - Microsoft Windows RPCSS DCOM Interface Denial of Service Vulnerability (2003 bid:8234 xfocus guys)

MS RPC Services (2)

- Microsoft RPCSS DCOM Interface Long Filename Heap Corruption Vulnerability (2003 bid:8459 nsfocus)
- Microsoft RPCSS DCERPC DCOM Object Activation Packet Length Heap Corruption Vulnerability (2003 bid:5458 eeye, nsfocus etc..)
- Microsoft Windows RPCSS Multi-thread Race Condition Vulnerability (2003 bid:8811 iss)

Information from EndPoint Mapper

- DEFAULT: Open For ALL (135, 139, 445, 1025, 1026)
- rpcdump /s ip /l
- Registered Endpoint(system service)
- Endpoint Type
- /l Options (Access)
- Samples

```
ncadg_ip_udp(Datagram (connectionless) UDP/IP)
xxx.xxx.xxx.xxx[1026] [32d90706-b698-4029-b236-e18ebff582b1] :YES
```

Information – Active Detection

- SPIKE and rpcenum.idc
- BIND -> BIND_ACK (With Provider Rejection Error)
- A tiny program (rpcdetect.c for SPIKE)
- Samples

```
./rpcdetect xxx.xx.xx.xxx 135 e1af8308-5d1f-11c9-91a4-08002b14a03a  
BIND Send OK  
BIND_ACK receive  
Provider rejection  
Abstract syntax not supported
```

```
./rpcdetect xxx.xx.xx.xxx 135 e1af8308-5d1f-11c9-91a4-08002b14a0fa  
BIND Send OK  
BIND_ACK receive
```

- <http://razor.bindview.com/tools/desc/rpctools1.0-readme.html>
- http://www.hsc.fr/ressources/articles/win_net_srv/

Dave Aitel's SPIKE – msrpcfuzz

Advantage

- Fuzz X(string, integer...) library.
- SPK script(Test case seperated).
- Test Executor
- Various Protocol(MS RPC, SUN RPC, SMB ...)

Disadvantage

- Method (blind, fuzz, lots of useless test case)
- Exceptional Elements
- Network and *nix Host Only.

Acknowledgement

- **Halvar Flake** (Reverse Engineer Tools and Bugscam)
- **Dave Aitel** (SPIKE)
- **yuange** (Leading me to the Way of Automatic Security Testing)
- Inside Windows 3rd Edition
- **YongQ** (L^AT_EX and prosper)
- **Members of SST** (lovely guys)
- **OYXin, BladeSatan, dudu...** (Friends)

•
•
•

Thanks!

Questions?