Call the plumber –
You have a leak in your (named) pipe
Agenda

• Presenter introduction
• Key terms
• Connecting to named pipes
• Pipe ACLs And Connection Limitation
• Named pipes in the wild

• Enumerating And Scanning For Named Pipes
• Sniffing Named Pipes Content
• Fuzzing Named Pipes
• Exploitation And Impact
• Case studies & Live demo!
• Mitigation And Defense
Your host

Gil Cohen
CTO, Comsec Global

• IDF Programming course graduate (“Mamram”) and former waterfall developers

• Cyber Security professional with more than 12 years of experience

• Vast comprehensive knowledge in penetration tests, secured design, programmers’ training and information security in general

30 years
Established in 1987, Comsec has nearly three-decades of experience in all aspects of information security.

150 consultants
Allows us to deliver a broad spectrum of services and to provide a uniquely flexible service level.

600 clients
From blue chip companies to start-ups, Comsec has a deep sector expertise in most verticals and unparalleled understanding of our clients’ business environment.

22 countries
With offices in London, Rotterdam and excellence center in Tel Aviv, Comsec is able to deliver global impact through local presence spanning over 22 countries and five continents.
core Services

Innovation, Knowledge & Experience to Keep You Ahead of the Curve.
Key Terms
Introduction To Key Terms

**IPC** or **Inter-Process Communication**

- An operating system mechanism that allows processes and applications to manage shared data and communicate
- Categorized as clients and servers, where the client requests data and the server responds to client requests
- Many applications are both clients and servers, as commonly seen in distributed computing
Introduction To Key Terms

*Windows Named Pipes*

- One of the methods to perform *IPC* in *Microsoft Windows*

- **One-way** or **duplex** pipe for communication between the **pipe server** and one or more **pipe clients**

- Utilizes a unique file system called **NPFS**(Named Pipe Filesystem)

- **Any process** can **access** named pipes, subject to **security** checks

- **All instances** of a named pipe share the **same pipe name**, but each instance has its own buffers and handles
Introduction To Key Terms

**Windows Named Pipes**

Many configurations and variations:

- Half Duplex or Full Duplex.
- Byte-Oriented or Packet-Oriented.
- Local or **Network.**

Inter-process communication is not only local!

Named pipes network communication is **not encrypted** and uses the protocols **SMB (port 445) or DCE\RPC (port 135)**
Introduction To Key Terms

**RPC or Remote Procedure Call**
- A protocol that allows one program to invoke a service from a program located on another computer
- No need to understand the network's structure\details
- Uses port 135 TCP or UDP

**DCE/RPC or Distributed Computing Environment / Remote Procedure Calls**
- A facility for calling a procedure on a remote as if it were a local procedure call
- To the programmer, a remote call looks like a local call
Introduction To Key Terms

**SMB or Server Message Block**
- An application-layer network protocol providing shared access to files, printers, serial ports etc.

- Mostly used for file sharing
  - \192.168.1.1\c\$\Users\manager\Documents
  - \fileserv\public\shareddocs

- Also provides an authenticated inter-process communication mechanism

- Uses port number 445 TCP
Introduction To Key Terms

Named and Unnamed \ anonymous Pipes

Two types of named pipes:
• **Named pipes**: has a specific name, all instances share the name

• **Unnamed \ anonymous pipe**: is not given a name
  o Only used for communication between a child and it’s parent process
  o Always local; they **cannot be used** for communication over a network
  o **Vanishes** as soon as it is **closed**, or one of the process (parent or child) completes execution
  o Actually named pipes with a **random name**
Connecting To A Named Pipe
Connecting To A Named Pipe

• All pipes placed in the root directory of NPFS

• **Cannot** be mounted within the normal filesystem

• Mounted under the special path - `\\\pipe\{pipe name}`
  o A pipe named "foo" would have a full path name of:
    `\\\pipe\foo`
  o Remote connection:
    `\\10.0.0.1\pipe\foo`

• Can be connected to programmatically or with dedicated tools
Connecting To A Named Pipe

**IO Ninja**

- Named pipes (and other communications) Swiss army knife
  - [http://tibbo.com/ninja.htm](http://tibbo.com/ninja.htm)
- Free for non-commercial usage 😊
Connecting To A Named Pipe

- This is how it looks in **Wireshark** (SMB communication)
Pipe ACLs And Connection Limitation
Pipe ACLs And Connection Limitation

- Named pipes are implemented by a filesystem driver in Windows NT, npfs.sys, which supports security descriptors.

- Security descriptors are used to control access to named pipes.

- By default DACL (Discretionary Access Control Lists) permissions are set to everyone using anonymous login (null sessions).

- ACLs can be modified to allow only specific users (same as file ACLs).
Named Pipes have Access Control Lists.
For the following pipe it is permitted to everyone to connect:

```
G:\Network\Named Pipes>pipeacl \??\pipe\initshutdown
Revision: 1
Reserved: 0
Control: 8004
Owner: BUILTIN\Administrators (S-1-5-32-544)
Group: SYSTEM (S-1-5-18)
Sacl: Not present
Dacl: 3 aces
(A) (00) 0012019b : Everyone (S-1-1-0)
(A) (00) 0012019b : Anonymous (S-1-5-7)
(A) (00) 001f01ff : BUILTIN\Administrators (S-1-5-32-544)
```
Pipe ACLs And Connection Limitation

Named pipes ACLs enumeration

- Using other 3rd party tools
- For example: Beyond Security Pipe Security Editor

An old utility, deprecated

Win32 Pipe Security Editor for Windows NT/2000/XP
http://retired.beyondlogic.org/solutions/pipesec/pipesec.htm
Another limitation of Windows Named Pipes is the **max number of instances** of a pipe.

<table>
<thead>
<tr>
<th>Pipe Name</th>
<th>Instances</th>
<th>Max Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>InitShutdown</td>
<td>3</td>
<td>-1</td>
</tr>
<tr>
<td>lsass</td>
<td>4</td>
<td>-1</td>
</tr>
<tr>
<td>ntsvcs</td>
<td>3</td>
<td>-1</td>
</tr>
<tr>
<td>scerpc</td>
<td>3</td>
<td>-1</td>
</tr>
<tr>
<td>Winssock2\CatalogChangeListener-38c-0</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>epmapper</td>
<td>3</td>
<td>-1</td>
</tr>
<tr>
<td>Winssock2\CatalogChangeListener-2ac-0</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>LSM_API_service</td>
<td>3</td>
<td>-1</td>
</tr>
<tr>
<td>eventlog</td>
<td>3</td>
<td>-1</td>
</tr>
<tr>
<td>Winssock2\CatalogChangeListener-290-0</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>atsvc</td>
<td>3</td>
<td>-1</td>
</tr>
<tr>
<td>Winssock2\CatalogChangeListener-2a8-0</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>spoolss</td>
<td>3</td>
<td>-1</td>
</tr>
<tr>
<td>Winssock2\CatalogChangeListener-658-0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>wkssvc</td>
<td>4</td>
<td>-1</td>
</tr>
<tr>
<td>Winssock2\CatalogChangeListener-314-0</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>ma_d5599bbe-4623-46a0-98a0-fa5e985813e2_DC80000004FBAE5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>ma_d5599bbe-4623-46a0-98a0-fa5e985813e2_6360000001DDBBB</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>ma_5bd9fa52-9d71-e8fd-2b0b-306ab91d3db1_2052.000000000C9E120</td>
<td>-1</td>
<td></td>
</tr>
<tr>
<td>mmsserver</td>
<td>5</td>
<td>-1</td>
</tr>
<tr>
<td>mfvtp_mfemms_listenerpipe</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Named pipes in the wild
Conficker case study

• Conficker is a computer worm targeting the Microsoft Windows operating system that was first detected in November 2008.

• It uses flaws in Windows OS software and dictionary attacks on administrator passwords to propagate while forming a botnet.

• It has been unusually difficult to counter because of its combined use of many advanced malware techniques.

• It infected millions of computers including government, business and home computers in over 190 countries (!).
Conficker case study

Worm: Win32 Conficker

Computers within a network that have weak passwords and without latest security update/anti-virus softwares are infected with the worm.

Computers that have unsecured/open shared folders without latest security update/anti-virus softwares are infected with the worm.

Computer with strong password, secured shared folder, latest security update and anti-virus software is protected from the worm.
Conficker case study

• Variant C creates a **named pipe**, over which it can **push** URLs for downloadable payloads to other infected hosts on a local area network.

• Named pipes can be used for C&C purposes!

• Used in other Trojans as well: Moker, ZxShell and even Petya uses it to transfer extracted passwords.
Enumerating And Scanning For Named Pipes
Named pipes can be enumerated using different testing tools. For locally detecting which named pipes are opened, it is possible to use Sysinternals’ pipelist:

https://download.sysinternals.com/files/PipeList.zip
Enumerating And Scanning For Named Pipes

Named pipes ACLs enumeration
using SysInternals’ pipeacl
• enables viewing permission of a certain named pipes:

C:\> pipeacl \pipe\lsarpc

Revision: 1
Reserved: 0
Control: 8004
Owner: BUILTIN\Administrators (S-1-5-32-544)
Group: SYSTEM (S-1-5-18)
Sacl: Not present
Dacl: 3 aces
(A) (00) 001f01ff : BUILTIN\Administrators (S-1-5-32-544)
(A) (00) 0012019b : Anonymous (S-1-5-7)
(A) (00) 0012019b : Everyone (S-1-1-0)

www.securityfocus.com/tools/2629
Enumerating And Scanning For Named Pipes

Forgotten Metasploit module called **Pipe auditor** enumerate **remotely** accessible named pipes, over SMB (**Pipe_Auditor**) or RPC (**Pipe_dcerpc_auditor**)
Sniffing Named Pipes Content
Sniffing Named Pipes Content

IO Ninja also enables sniffing and monitoring traffic of a chosen named pipe:

http://tibbo.com/ninja.html
Fuzzing Named Pipes
• Fuzzing or fuzz testing is an automated software testing technique that involves providing invalid, unexpected, or random data as inputs to a computer program.

• Done with fuzzers – automatic fuzzing tools

• The program is then monitored for exceptions such as crashes and potential RCEs.

• Typically, fuzzers are used to test programs that take structured inputs.
Fuzzing

Two types of fuzzing approaches:

**Dumb (“Black Box”)**
- Go over all possible inputs without understanding the expected ones (sometimes implemented using random data)
- Simple to implement, sometimes impossible to execute using the sequential approach

**Smart (“White Box”)**
- Understand the expected input and fuzz along the edges (mix expected data template with random values)
  - Smart data generation
- Harder to implement, more code coverage
Fuzzing Named Pipes

Windows IPC Fuzzing - dump-fuzzing named pipes script

Exploitation And Impact

• Many pieces of software work with hidden and/or undocumented APIs

• The forgotten nature of named pipes leave an uncharted territory of socket-like interfaces that can contain vulnerabilities

• Named pipes fall in between App PT and Infra PT.
  o App pentesters usually connects to typical app ports, RPC and SMB not included.
  o When Infra pentesters encounter RPC\SMB they try to gain credentials, not check for named pipes.

• If software reads data from the named pipe without any validation of the content, the attacker might trigger Buffer Overflow leading to Denial of Service of the software and even Remote Code Execution.
Exploitation And Impact

• If named pipe ACLs allow remote access, remote DoS or RCE can be triggered

• Research of the cause behind the crash will allow the attacker to facilitate it as a zero day vulnerability

• Could be used to spread a malware in an internal network, as recently seen in the WannaCry ransomware campaign

GAME OVER
Case study: Viber, qBittorrent, SugarSync
Viber, qBittorrent & SugarSync case study

Viber
• Cellular & endpoint social communication
• Free calls, text and picture sharing with anyone
• Competitors of WhatsApp
• 800 million users worldwide
Viber, qBittorrent & SugarSync case study

**qBittorrent**
- a cross-platform client for the BitTorrent protocol
- Free and open-source, released under the GPLv2
- Written in C++

**SugarSync**
- A cloud service that enables active synchronization of files across computers and other devices
- Used for file backup, access, syncing, and sharing
- Supports variety of operating systems, such as Android, iOS, Mac OS X, and Windows devices
Exploitation And Impact

The applications use the widely used **QT framework**:

• A cross-platform application development framework for desktop, embedded and mobile. Supports multiple platforms and operating systems

• The applications use the **qtsingleapp** functionality which is responsible for writing temp files

• By **fuzzing** the named pipe both locally and remotely, we managed to **remotely crash the programs** and in Qbitorrent, also a possible **remote command injection**
Demo
Mitigation and Defense
Mitigation And Defense

Developers point of view

Know the risk!

- When creating a named pipe, set a secured ACL to allow only authorized connections to the named pipes.

- Follow the **least privilege** approach
  - Giving a user account only those privileges which are essential to perform its intended function.

- If possible, limit the maximum number of instances of a named pipe, thus effectively limiting the number of simultaneous connections.
Mitigation And Defense

**Users\3rd party software clients point of view**

Know the risk!

- Block all unnecessary SMB and RPC services (ports 135 and 445), especially over WAN/Internet
- Segment the network according to security best practices
- Always install the latest software security patches
Hackers’ point of view
Know the opportunity!

• Well… Hack

• Explore remotely accessible named pipes and test for RCE and DoS whenever seeing open SMB or RPC ports

• Have fun! 😊
Closing remarks

• Windows named pipes are a forgotten, remotely accessible, socket-like interface

• A whole, newly rediscovered, potential world of local and remote vulnerabilities – increased attack surface

• Don’t ignore named pipes in Windows desktop applications

Stay safe
Thank you

Gil Cohen

twitter.com/Gilco83
www.linkedin.com/in/gilc83
Gilc@comsecglobal.com
www.comsecglobal.com

Gr33tz & Th2nkz:
Aviad Golan @AviadGolan, linkedin.com/in/aviadgolan
Peter Savranskiy - peters@comsecglobal.com
Reuvein Vinokurov - reveinv@comsecglobal.com
Coral Benita - coralb@comsecglobal.com
Meareg Hunegnaw - mearegh@comsecglobal.com
Roni Fenergi - ronif@comsecglobal.com
Sharon Ohayon - sharono@comsecglobal.com
Josh Grossman - joshg@comsecglobal.com