Rootkits and Trojans on your SAP Landscape

Ertunga Arsal

Chaos Communication Congress 2010

Agenda

- Introduction to Enterprise Security
- SAP* Applications in General
- BASIS (SAP infrastructure) Security
- Attacks to ABAP Programs
- ABAP Rootkits
- The Threat Agents
- How To Stay Secure

*SAP refers to SAP R/3 and Netweaver applications throughout this presentation, not the company.

About Me

- Ertunga Arsal
  - Security Researcher with focus on Enterprise Systems
  - Founder of ESNC GmbH, a company specialized in SAP Security
- Officially acknowledged for the following Security Patches :
  - SAP Note 1484692 - Protect read access to password hash tables
  - SAP Note 1497104 - Protect access to PSE
  - SAP Note 1421005 - Secure configuration of the message server
  - SAP Note 1483525 - New security center in SAP GUI 7.20
  - SAP Note 1485029 - Protect read access to key tables
  - SAP Note 1488406 - Handling the generated user TMSADM
  - SAP Note 1511107 - Executing freely determined code using transaction SE37
  - SAP Note 1510704 - Missing Authorization Check in AFX Workbench report

Typical Enterprise

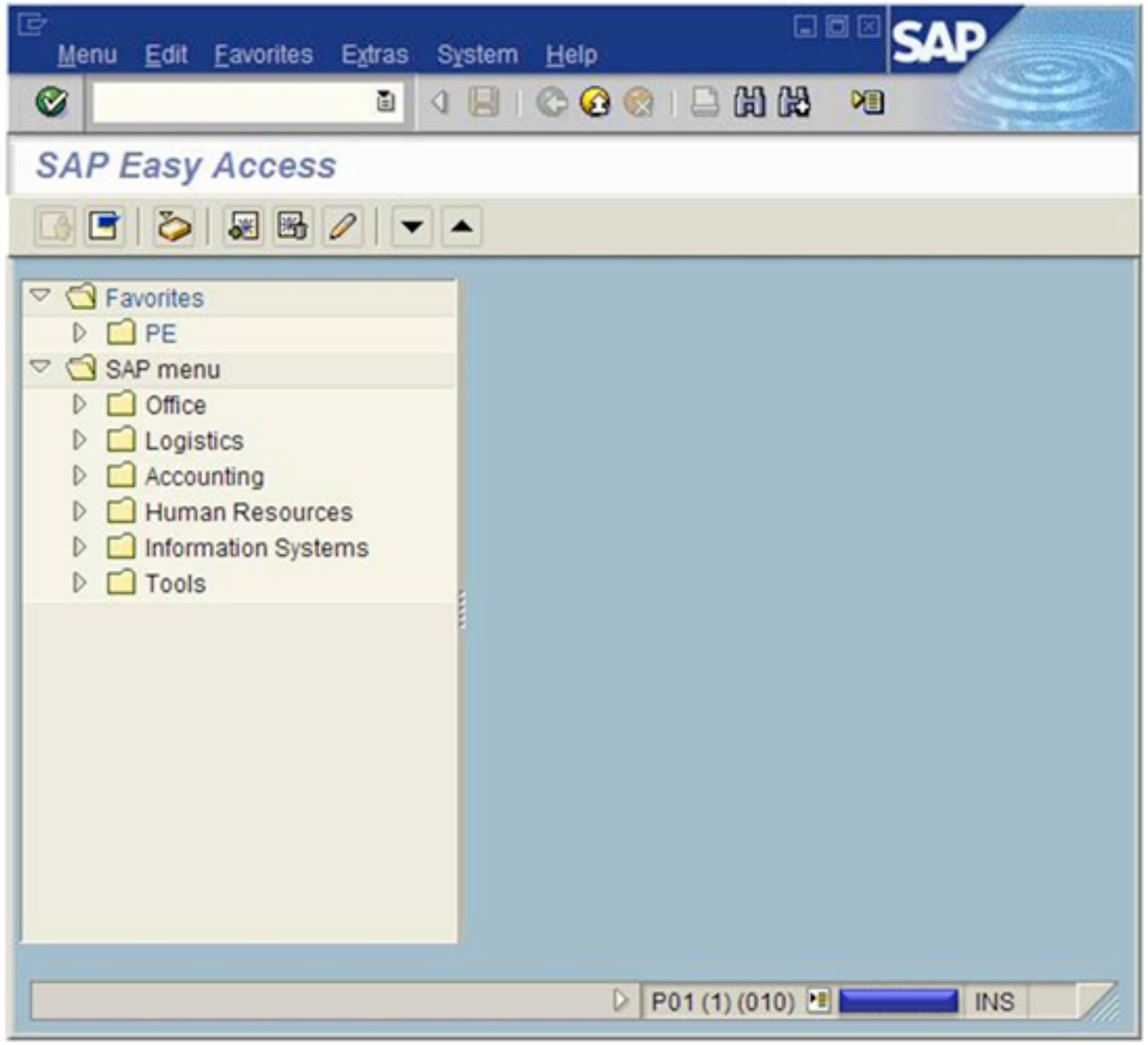
- Has more than a thousand of employees
- Is a circus of IT Systems
  - Mixture of operating systems, databases, applications
    - And their different versions
    - Usually implemented by different teams
    - Spanning to a lot of years
- Decision makers care more about their bonus than the interest of the company
- Is a political battlefield about who has the bigger balls [unisex term]

Typical Enterprise Security

- Even medium level of IT security is too expensive to achieve
  - Missing asset management (how many Oracle DBs, Windows servers, etc?)
  - Tons of security scanning, to few remediation chasing
  - Many of the vulnerabilities cannot be mitigated
- Obsessed by Cross Site Scripting
- IT security departments cannot influence security decisions of business applications much, because of political reasons
- Nobody cares about the hacked UNIX machine, SQL DB, or others
  - If they are not directly held responsible (CYAS - Cover Your Ass Security)
    - SoX, PCI-DSS, legal requirements, ...
- Defacements and similar security incidents are budget approvers

SAP Systems

- Business specific
  - HR, Finances, Logistics…
- Industry solutions
  - Defense & Aerospace, Oil & Gas, Banking, Chemicals...
- Hold the Crown Jewels
  - Hence “Business”
- Are usually extensively customized
  - SAP consultants on-site
  - Long running implementation projects
- Less exposure to typical hackers
  - Who would learn ABAP for hacking?
  - How would someone try it at home?



Sutton's Law

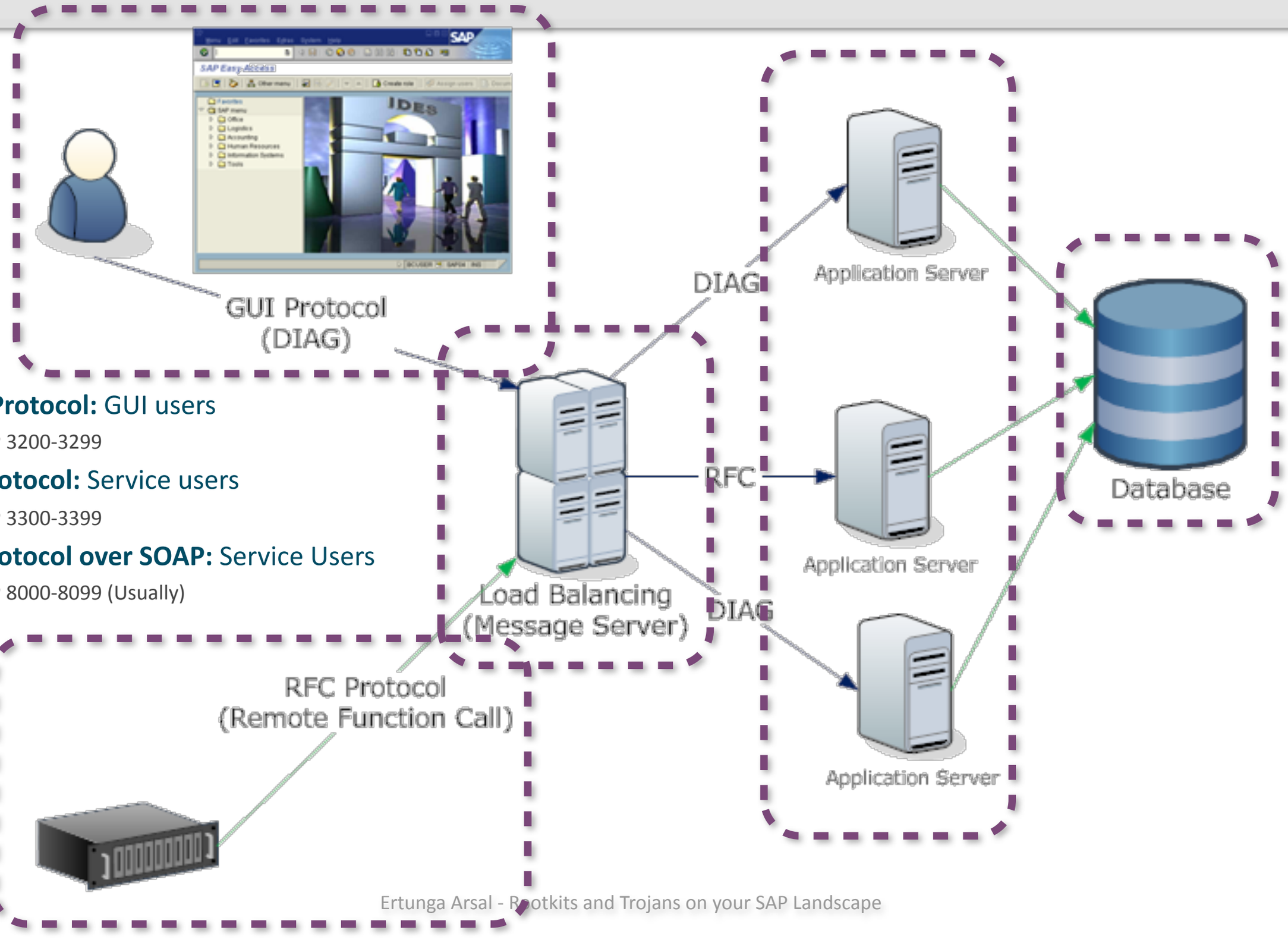
- Main principle: "When diagnosing, one should first consider the obvious"
- Named after a bank robber, Willie Sutton
  - Sutton was asked why he robbed the banks
  - His response*: "Because that's where the money is"

- Probably he never said this

SAP Security

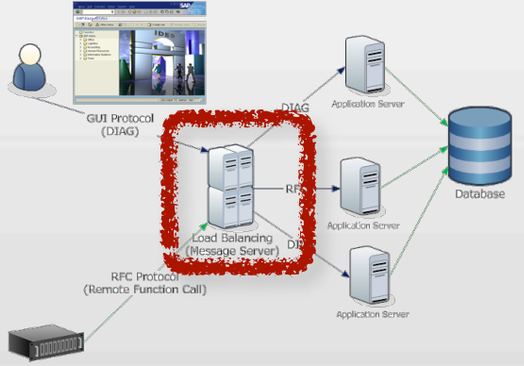
- Security mostly focuses on authorizations and segregation of duties
  - SOD's main focus is the actions of a single person
  - Two guys get together = throw away your SOD investments
  - Weak passwords (99% of the case) = throw away your SOD investments
- Intrusion prevention is still a baby
  - How many signatures does your expensive IDP have for business apps?
- Risks are underestimated/general IT Security efforts are typically unbalanced at companies
  - How many Global 500s are running SAP for the core business?
  - How many people from their IT Security teams have SAP security skills?
- Unlike e.g Active Directory, SAP systems belong to the business, not the IT
- Security departments usually fail when they are challenged
  - Either missing skills or "**This attack is too sophisticated, nobody can do it**" response

SAP: Simplified Connection Overview

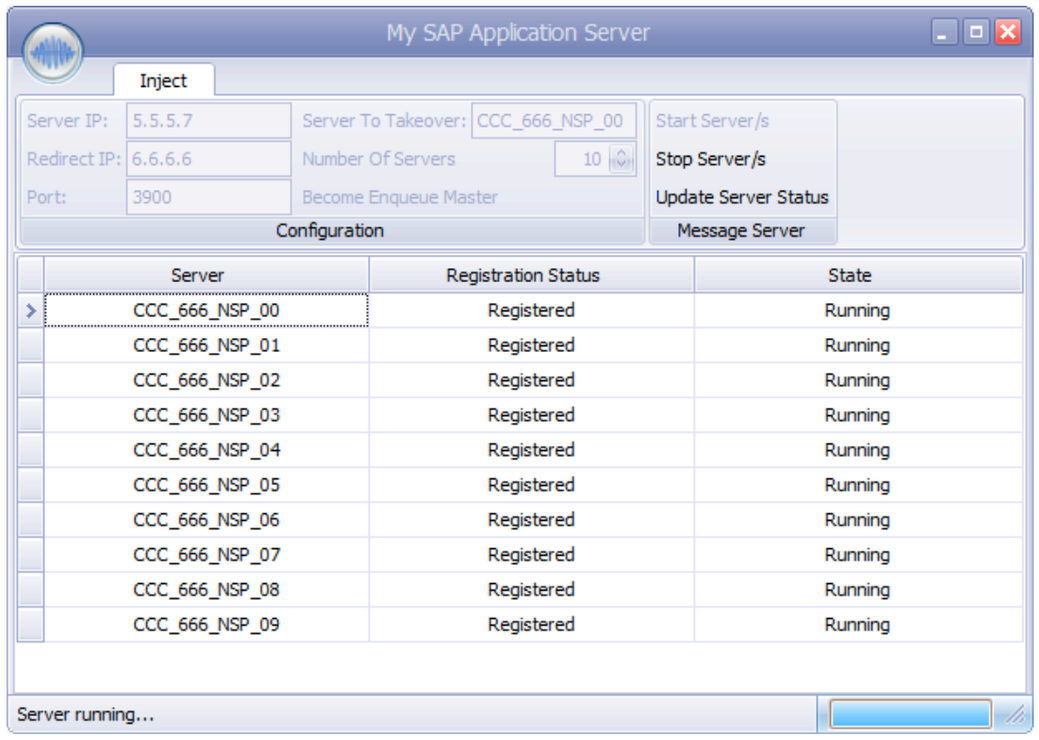


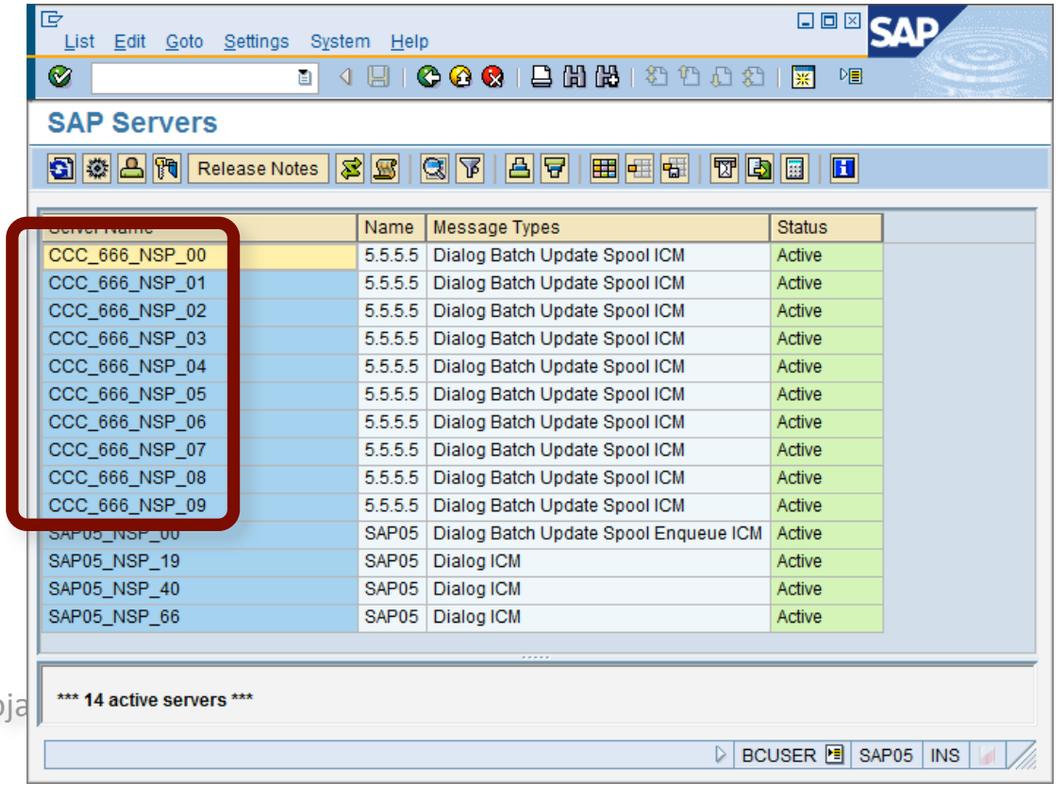
- **DIAG Protocol:** GUI users
  - TCP 3200-3299
- **RFC Protocol:** Service users
  - TCP 3300-3399
- **RFC Protocol over SOAP:** Service Users
  - TCP 8000-8099 (Usually)

SAP Load Balancer

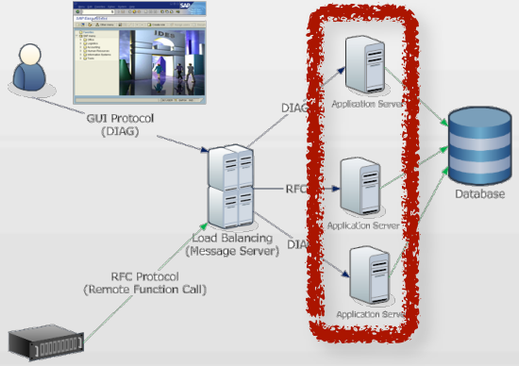


- "Message Server"
- If not properly configured, an attacker can register its own servers [top pic - PoC]
- Can fake the clients, MITM or more
  - Implement ms/acl_info access control to protect it!



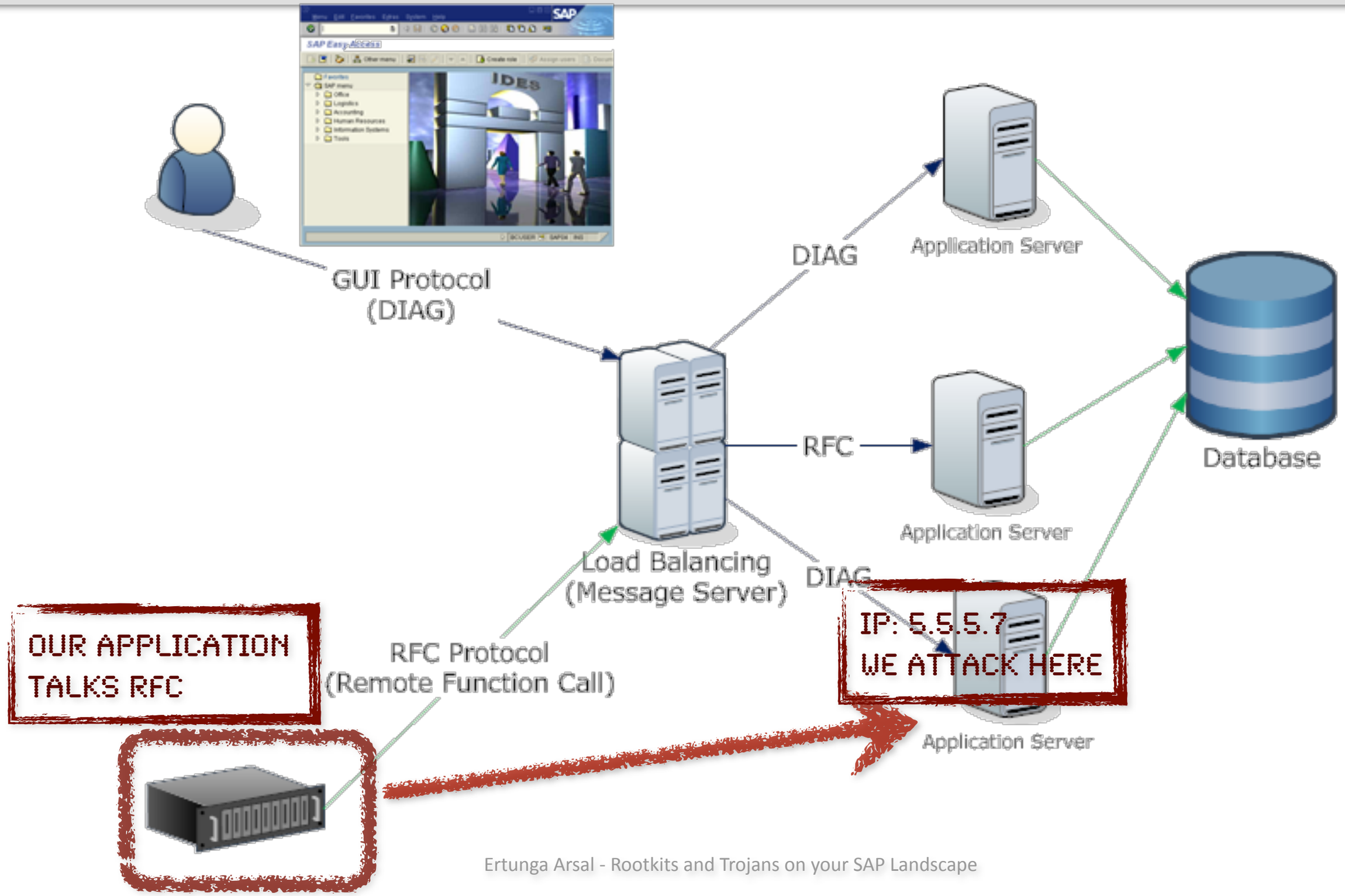


SAP Application Server



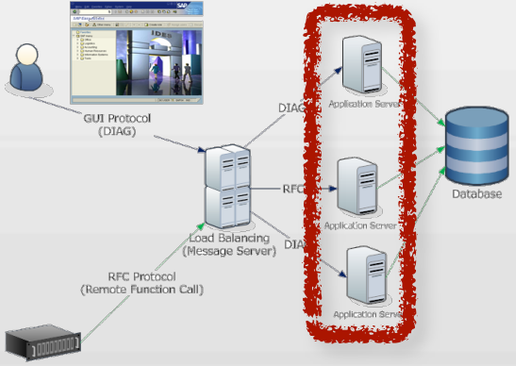
- Real name the "Gateway"
- Built-in remote shell functionality via RFC
  - –Good for remote administration without authentication
  - –Supports all operating systems (AIX, HP-UX, Z/OS, Win...)
  - –Can be restricted via secinfo ACL configuration
  - –Mariano mentioned this at BH in 2007
- Secinfo/reginfo can be bypassed with ease
  - –Make sure you apply the latest kernel security patches and you have a restrictive secinfo/reginfo configuration!

DEMO: Remote Shell



“GUI users are the most powerful users” myth

and RFC

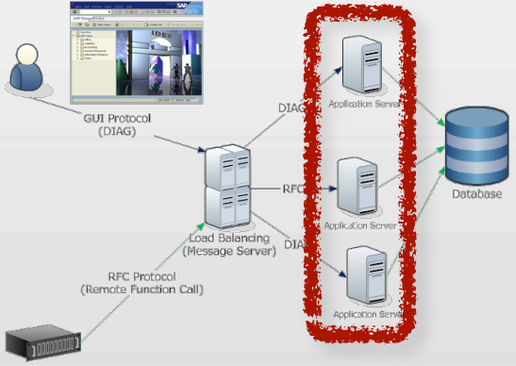


- RFC (Remote Function Call) protocol lets you run functions remotely
  - To run; use Java, C, etc. with RFC-SDK or simply execute the test program **startrfc**. Following creates a new user with god rights:

    startrfc -3 -h 10.1.5.4 -s 05 -c 010 -u ERTUNGA -p CCC42 -F SUSR_RFC_USER_INTERFACE -E USER=SATRIANI -E ACTIVITY=01 -E PASSWORD=RUBINA -E USER_TYPE=A -T USER_PROFILES, 12,r=-<press ENTER>**SAP_ALL**<press enter> <press ctrl-z and enter>
- There is no exploit involved. Everything is intended functionality.
  - Beats “RFC users are not a threat because they cannot login via SAPGUI”
  - Time to recheck company’s shared folders and eliminate hardcoded passwords.
- RFC (a.k.a communication) users are thus very very important!
  - Secure their passwords and make them part of the password change process
  - Don’t forget: GUI (dialog) users which have S_RFC rights can also execute remotely
  - SAP_ALL FOR COMMUNICATION USERS IS A NO GO!

A Few RFC's to note down and protect:

(Proper user authorizations is the key)

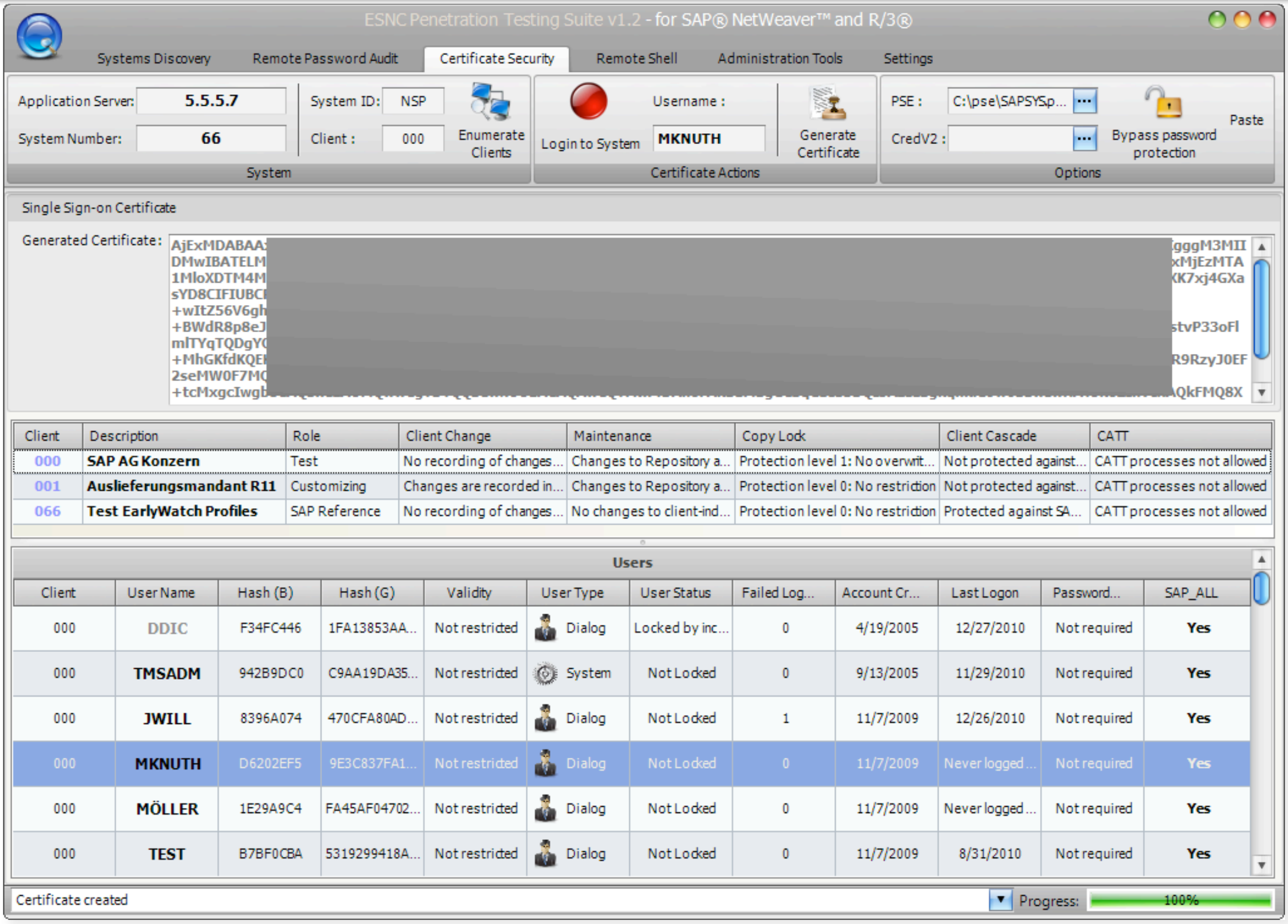


- RFC_READ_TABLE
  - Reads the contents of any table (Including ones with sensitive data e.g salary information)
  - Has bugs in converting e.g binary fields
    - 1 Byte = 2 Hex, so 20 byte hash -> 40 hex chars
    - Only returns first 20 chars because of miscalculation -> only first half of the password hashes
- SUSR_RFC_USER_INTERFACE
  - can be used for creating/modifying users.
- RFC_ABAP_INSTALL_AND_RUN
  - Takes ABAP source lines and executes them
    - does not execute on production systems but non-production does not mean that system is unimportant!
  - Widely known!!! tighten user authorizations to prevent abuse
  - More restricted in latest NetWeaver Systems
    - SAP_ALL RFC users don't have those restrictions!!!
- !!! RFC can be encapsulated in SOAP messages (SOAP RFC)
  - Company's internal proxy suddenly opens the doors to all SAP systems
  - Disable it if not used!

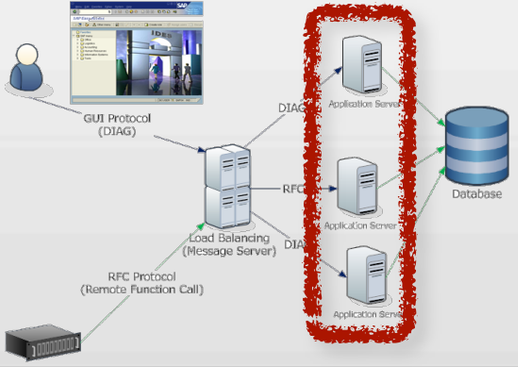
Single Sign-on (SSO2)

- Is a convenience feature, not a security feature
- RTFM: Secure Store and Forward [SSF] documentation
- Personal Security Environment files hold the private key data
  - Stored per default in **SAPSYS.pse** file or DB table **SSF_PSE_D**
- If an attacker obtains it, it can create authentication tickets for the victim system
  - Accepting these tickets is enabled per default
  - Attacker can logon as any user
- The idea of home brewed authentication tickets first came from an SAP guru: Ralf Nellessen

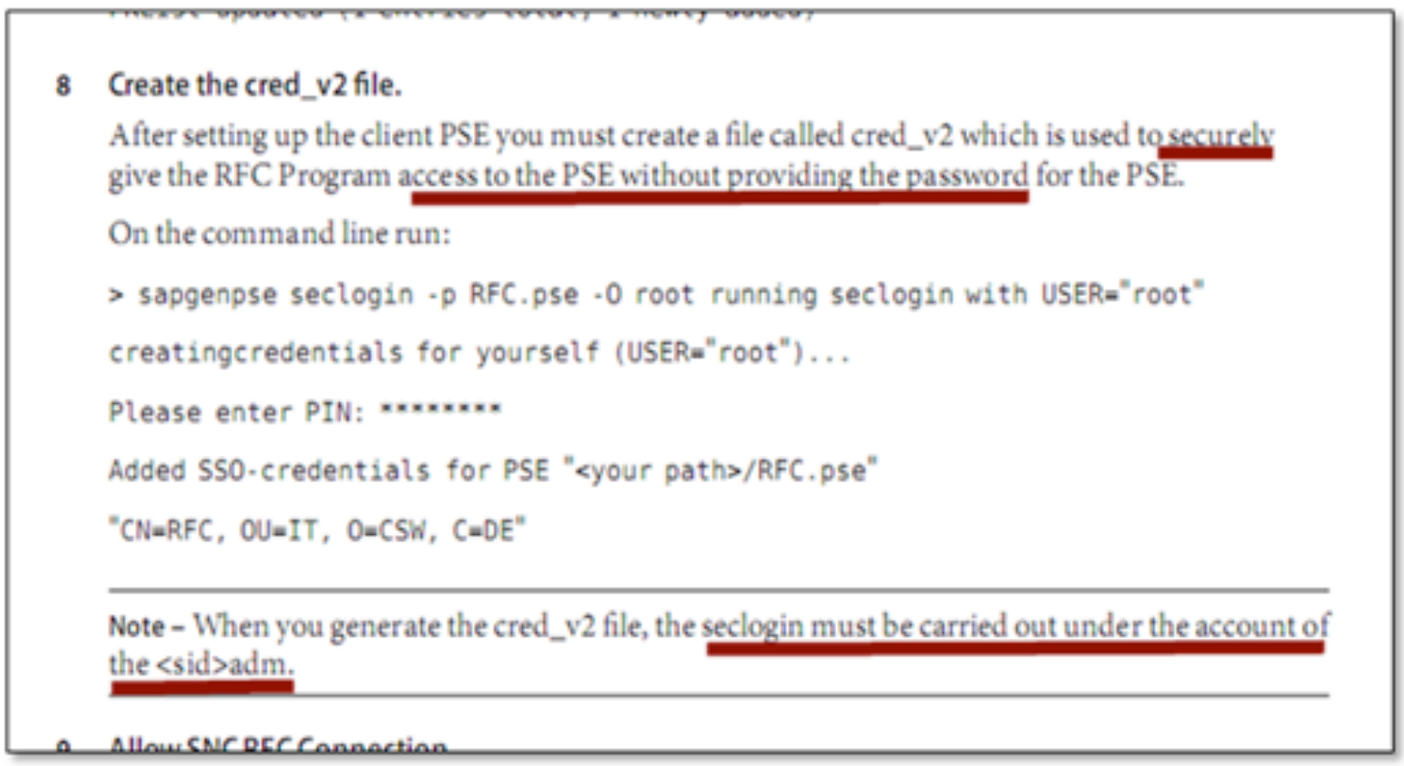
DEMO: Certificate Attacks



Single Sign-on (SSO2)



- The private key container (PSE) can be pin-protected
- I was trying to see whether the pin mechanism had any flaws
  - Found a way, so googled for more info
  - Somebody was unconsciously ahead and even documented that :)



8 Create the cred_v2 file.

After setting up the client PSE you must create a file called cred_v2 which is used to securely give the RFC Program access to the PSE without providing the password for the PSE.

On the command line run:

```
> sapgenpse seclogin -p RFC.pse -O root running seclogin with USER="root"
creatingcredentials for yourself (USER="root")...
Please enter PIN: ********
Added SSO-credentials for PSE "<your path>/RFC.pse"
"CN=RFC, OU=IT, O=CSW, C=DE"
```

Note – When you generate the cred_v2 file, the seclogin must be carried out under the account of the <sid>adm.

9 Allow SNC RFC Connection

Configuring Secure Network Communications for SAP

(http://dlc.sun.com/pdf/820-5064/820-5064.pdf)

- Disable accepting tickets using relevant profile parameters!

SAP Applications (ABAP)

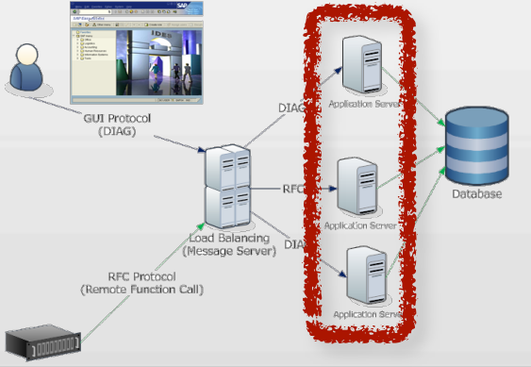
- ABAP code holds almost all of the business logic
- More than 2.000.000 programs are present at an SAP ECC 6.0 system after installation.
  - Some programs have more than 50.000 lines of source code
- ABAP Language is very powerful and easy to learn
  - High level and easy to read applications
  - Low level functionality is proxied to the kernel executables when required. e.g for encryption.
    - ABAP stack can “call” the kernel.
    - We’ll only focus on the native ABAP code for this presentation.

Dynamic ABAP

- Statement: `GENERATE SUBROUTINE POOL`
  - Dynamically generates ABAP code.
  - If the code is generated via user specified input, mistakes mean:
    - ABAP Injection
    - Game over
  - An example is the TMS_CI_START_SERVICE vulnerability

TMS_CI_START_SERVICE

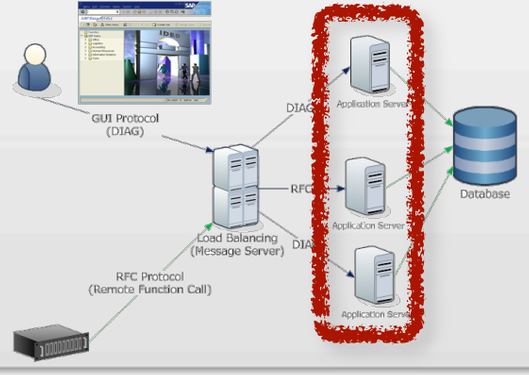
Executable Function



- Transport Management System required this
  - Transport == Software Installation
- It is an RFC
  - Remotely Executable Function Call
- Takes an input table as source code and if the parameters are specified properly, executes the contents of it.
  - Bingo!

TMS_CI_START_SERVICE

Executable Function



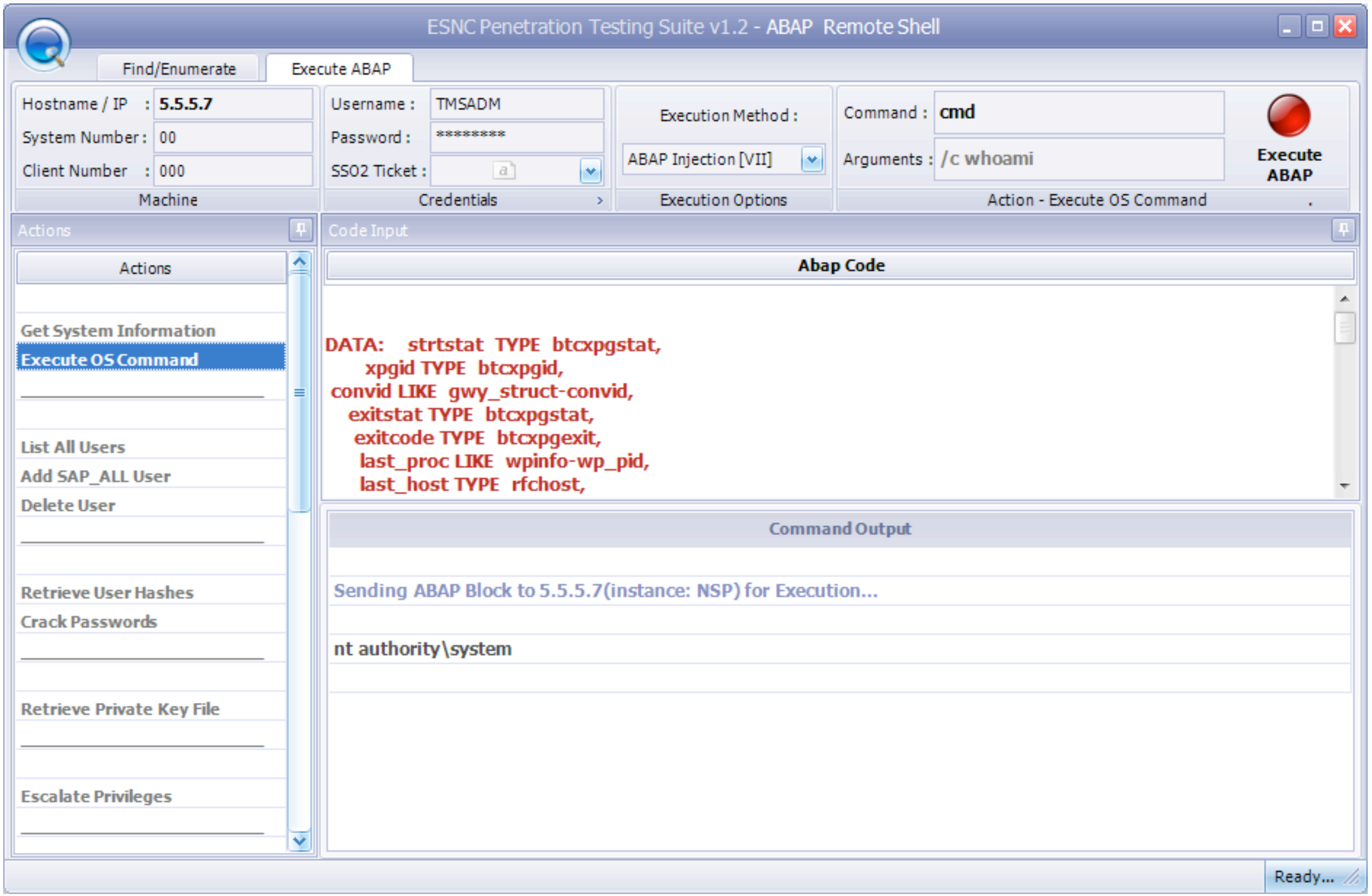
- Here is a simple representation of the vulnerable part of it:

```
Generate subroutine pool pp_table name ix_context.

perform (ix_command) in program (ix_context) tables pp_table.
```

- SAP patched it via:
  - SAP Note 1298160: Forbidden program execution possible
- TMSADM default password is at least for the last 5 years public
  - Password is “PASSWORD”

DEMO: ABAP Injection



SQL Injection

- ABAP typically uses parametrized queries.
  - Developers can still specify parts of sql statements dynamically by parentheses
- Not dynamic:
  - `SELECT ColumnA FROM TableA INTO[...]`
- Dynamic:
  - `SELECT(var_ColumName)FROM(var_TableName) INTO[...]WHERE(var_WhereClause)`
- Avoid dynamic statements where possible!

SQL Injection

- It's not a bug, its a feature in concept "Run Time Type Creation"
  - (e.g Z_RTTC report in NSP Test system)
  - https://wiki.sdn.sap.com/wiki/display/Snippets/Concept+of+Run+Time+Type+Creation
- Means generic table access - if not done properly
- !!! Also check the "EXEC SQL"
  - It allows DB specific dynamic queries

Cross Site Scripting

- Hard to believe we are still talking about it in 2011
- Proper sanitization/encoding of the input data is the key for self developed web code such as BSPs.
- If not done, an attacker can do everything related to XSS, plus steal e.g the SSO2 (Authentication) cookies from the clients
  - SSO2 cookies are stateless so client impersonation is a breeze.
    - Avoid using this mechanism without proper controls
  - If you have F5's or similar devices, encrypt cookies based on origin ip
    - can kill business if you encrypt based on full ip (32 bits)
    - can be too open if you just encrypt /24 of that ip
    - What happens to NAT clients, Firesheep?

ABAP Executable Manipulation

- Statement: `INSERT REPORT`
- Writes custom code to any ABAP program
- It's even possible to call an editor to make it more user friendly
  - Called editor is similar to the ABAP development environment
- Very suspicious if found in self developed code

RS_REPAIR_SOURCE

Executable Program

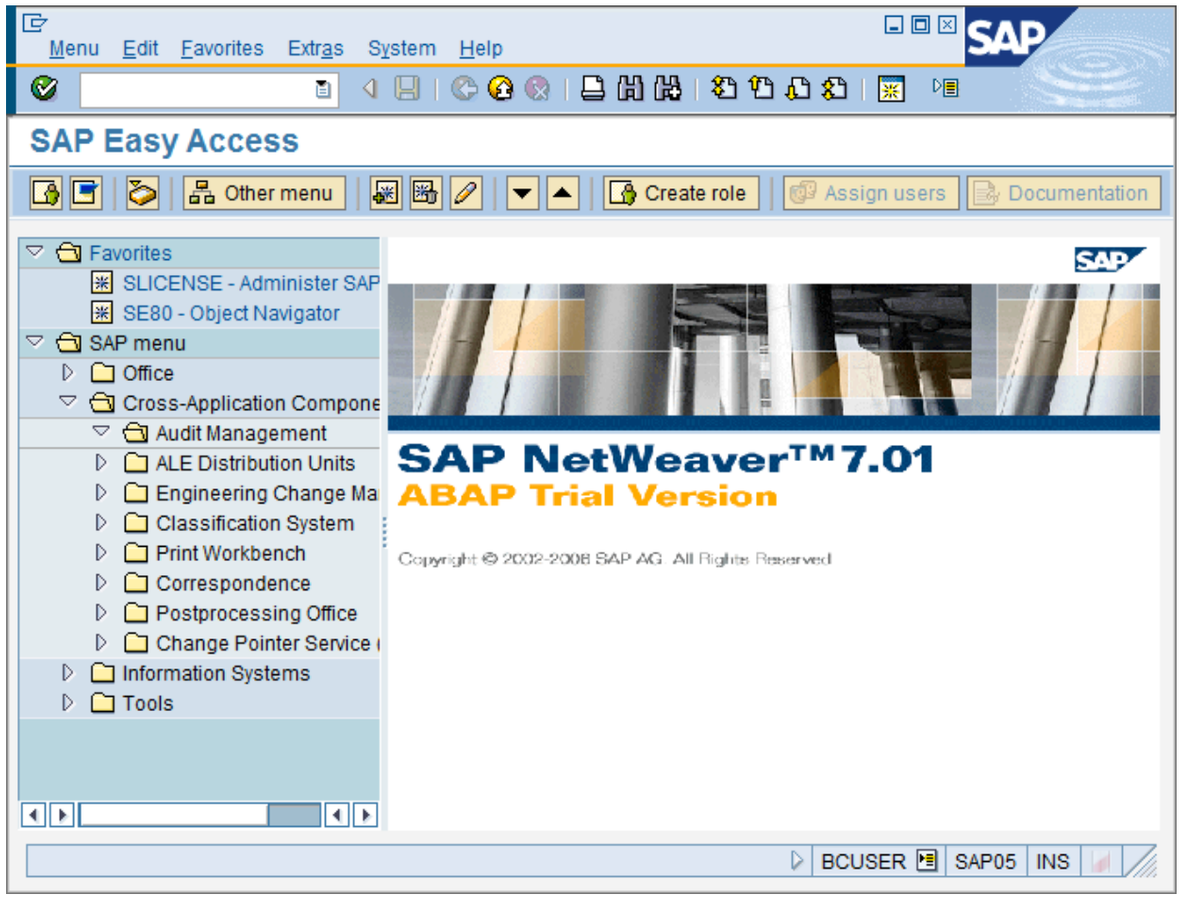
- Unpatched version does not have authorization checking.
- People with e.g SE38 rights can execute this and manipulate the system and data of it.
- Same as ABAP injection, only more convenient.
- SAP patched it via:
  - SAP Note 1167258: Program RS_REPAIR_SOURCE
- There are many other critical ABAP statements but they are beyond our scope for today. [one hour time limit hit]

ABAP Rootkits

- So, it is possible to modify system executables (ABAPs)
- An attacker can easily infect important ones executables and install an ABAP rootkit
- SAP has RFC functions that do not require user authentication by default (SRFC Function Group). This could be one candidate.
- Installed rootkit can give anonymous access to the attacker with functionality such as:
  - Installing SAP_ALL users
  - Manipulating ABAP reports
  - Running OS commands
  - Stealing hashes or PSE files
  - Deleting Logs

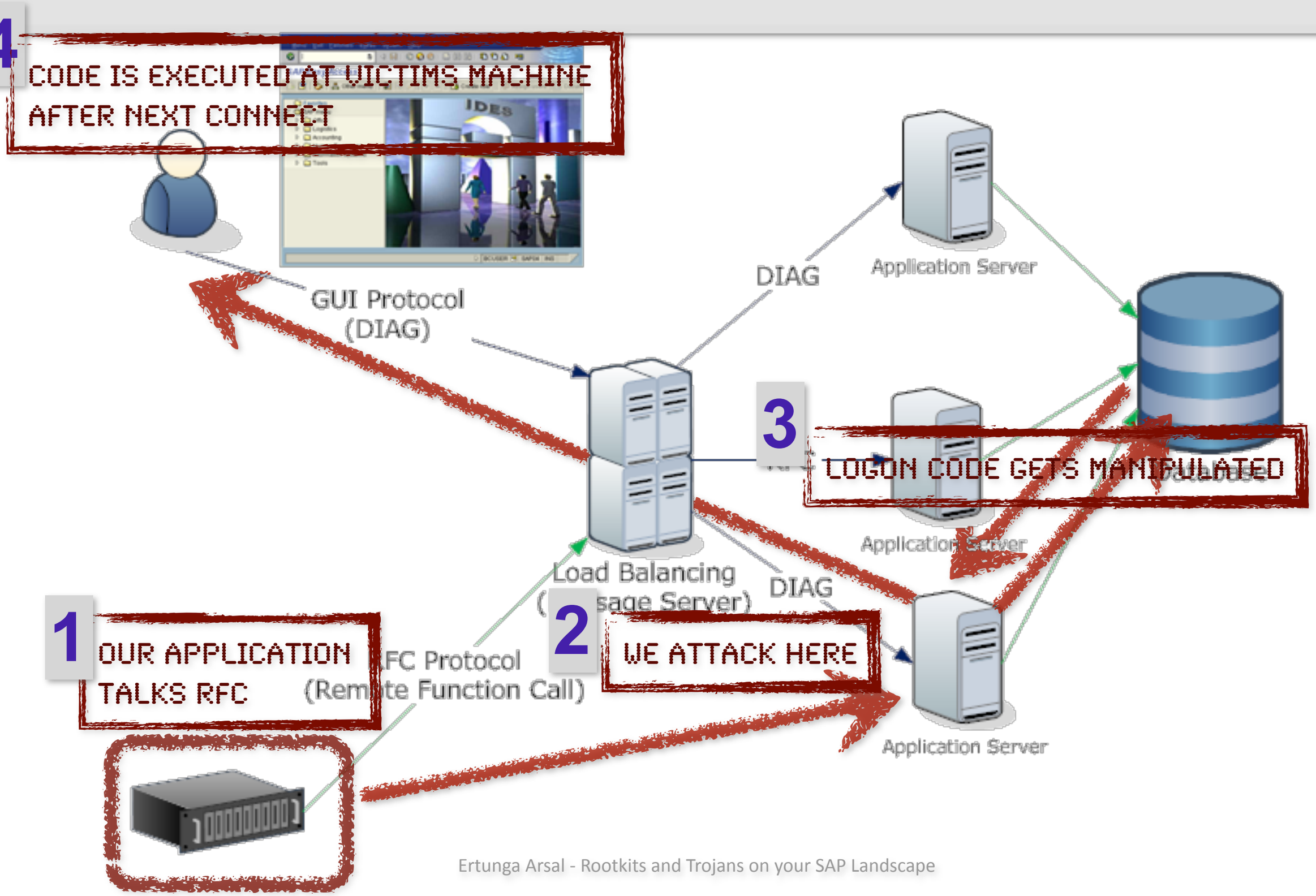
The Front End: SAPGUI

- Main application for SAP systems
- Runs on different platforms
- Has powerful features
- Has an API for client actions
  - Downloading
  - Uploading
  - Execute
  - Registry Access
  - etc.

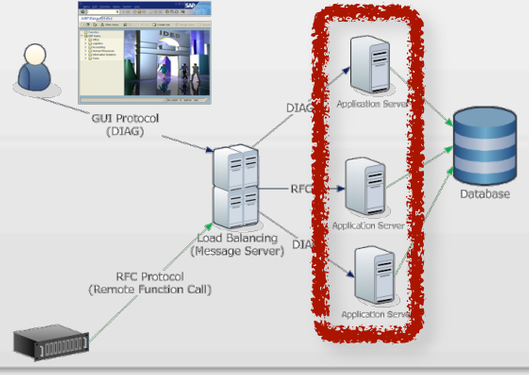


- With SAPGUI 7.20, there is a “Security Center” where certain actions can be blocked with an ACL

DEMO: Executing code on the client

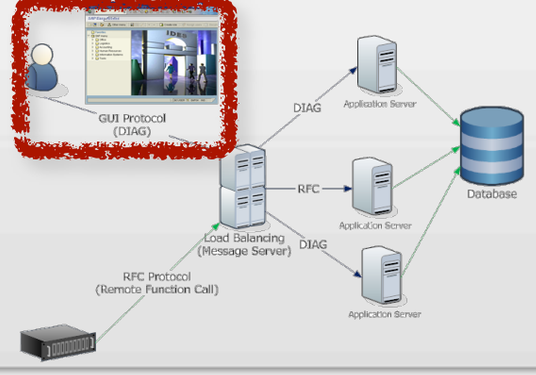


Triple-Penetration Attacks



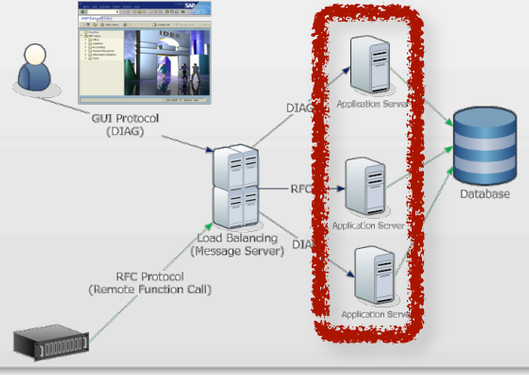
- **Penetration 1:** Attacker exploits the weakest system
  - Typical enterprise setup :
    - Testing/Development -> Quality Assurance -> Production
  - Among them, most unprotected are test/development systems
    - Who connect to these systems? Usually, admins and developers
  - TAGS: Password Security, Protection of the PSE files, Message Server Security, Database Security, OS Security, Network Sniffing, Missing Patches etc...

Triple-Penetration Attacks



- **Penetration 2:** Attacker infects clients which connect to the weakest system
  - Starts with modification/infection of the critical areas such as logon screen ABAP code
  - When admins/developers successfully login, malicious payload is downloaded and executed on these users' computers
    - Antivirus bypass, user mode rootkits, etc.
    - Sniffing SAP credentials e.g by tampering saplogon.ini

Triple-Penetration Attacks



- **Penetration 3:** Victim infects all the systems it later connects to
  - Modification of critical components of the newly accessed SAP systems
  - Internal production systems
  - Partner systems or other critical systems

0wn Half the W0rld's T0p Businesses

- Especially when initial target is an SAP Hosting or Training provider
  - Attacker pays a small amount to get a test account
  - Infects the system
  - Sits down and waits for the admin or other users to spread the infection to the systems they connect to

- Configure your SAPGUI security settings and avoid shared SAP systems where possible!
- Protect your end users via proper endpoint protection!

The Robin Hood W0rm for Fun and Profit

- Worm can access to the financial applications and data!
  - Sort of the "Worm writer's wet dream"
- Checks the balance at the year end closing
- If the company has profit:
  - **Donates** %0.01 of that amount to Red Cross, Red Crescent [put your favorite red organization here], SaveTheChildren or Wikileaks
- If infected systems contain HR systems:
  - Worm **publishes salary information** of the employees online
    - Tens of thousands of people notice that the jerk from department X gets twice as much money
    - Also consider the legal implications on the businesses

The Threat Agent: ABAP Developer

- Writes code that runs at the heart of the system
- The user rights and permissions don’t apply to him
- He can assign god rights to itself via code
  - Audit logs are typically disabled on development systems
    - If enabled, most probably developers will be able to disable/tamper them
    - remember to always log to an external system.
- You need to trust the developers more than your security team
  - Would you hire an ABAP developer who recently worked at a competitor?
    - IF answer EQUALS ”HELL, YEAH”, think again now.
  - How about the contracted ones that also provide services to other companies at the same time?

The Threat Agent: Dark Organisations

- STUXNET is very popular but…
  - SAP software is used for production of fighter jets, running power grids, oil & gas, critical production systems and more. Especially production, materials management, logistics and financials applications…
    - http://www.sap.com/industries/
  - Has much better API and documentation than PLCs and Step7
- Compared to the effort spent for STUXNET, it would be unreasonable to think that similar is not already done for such systems
  - What happens when you order wrong materials for the next Eurofighter aircraft?
  - How would you detect it?

How to stay secure?

(some more tips)

- Proper systems architecture is a prerequisite.
  - Read and Apply the “SECURE CONFIGURATION SAP NETWEAVER - APPLICATION SERVER ABAP” document from SAP
    - Make sure relevant people in your company also read it!
    - Check: https://service.sap.com/~sapidb/011000358700000968282010E.pdf
- Implement secinfo/reginfo and ms_aclinfo ACLs before system is first online
- Analyze your systems or use an ABAP integrity checking tool for detecting malicious system tampering and rootkit infections.
  - Currently only two products known to me. From Onapsis and ESNC GmbH
- Never give the development systems write permissions to the production systems’ transport import folders

How to stay secure?

- Have proper “check-in” and “leavers process” that take the ABAP developer risks into consideration
  - e.g. Full user password resets on certain development systems or other precautions when a developer leaves the company
  - Also consider putting external consultants in the scope
- Audit the code against security vulnerabilities before transporting to production systems
  - Currently only 2 automation products known to me. From ESNC GmbH and from VirtualForge GmbH
- Syncing passwords to development systems means, possibility of developers to capture valid passwords for production systems. Avoid it!

How to stay secure?

- Get rid of insecure and/or default passwords
- Disable backwards compatibility of passwords
- Follow vendor's security notes and guidelines
  - https://service.sap.com/securitynotes
- Convince the upper management that staying 2 years behind the security patches is a bad idea!
- Install the latest security patches
- Install the latest security patches
- Install the latest security patches

Credits/Thanks

- Stefan Fuenfrocken from EUROSEC
- Ralf Nellessen from TRUSTWERK
- Christian Wippermann from SAP
- Everyone @ Product Security Response Team/SAP

Questions?

Ertunga Arsal

ertunga at_sabanciuniv.edu

This publication contains references to products of SAP AG. SAP, ABAP, SAPGUI and other named SAP products and associated logos are brand names or registered trademarks of SAP AG in Germany and other countries in the world. SAP AG is neither the author nor the publisher of this publication and is not responsible for its content.

This presentation and the accompanying paper is for educational purposes only, I will not be held responsible for what you do with this information, you use it at your own risk.