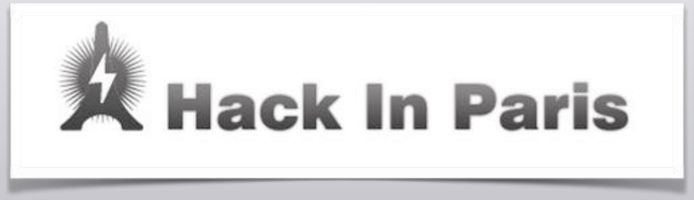


SAP Security: Real-life Attacks to Business Processes

Ertunga Arsal
ertunga@esnc.de

Agenda

- Business Processes
- SAP Systems
- Exploit Demo
- External Payment Solutions on SAP
- How to Stay Secure
- About Us

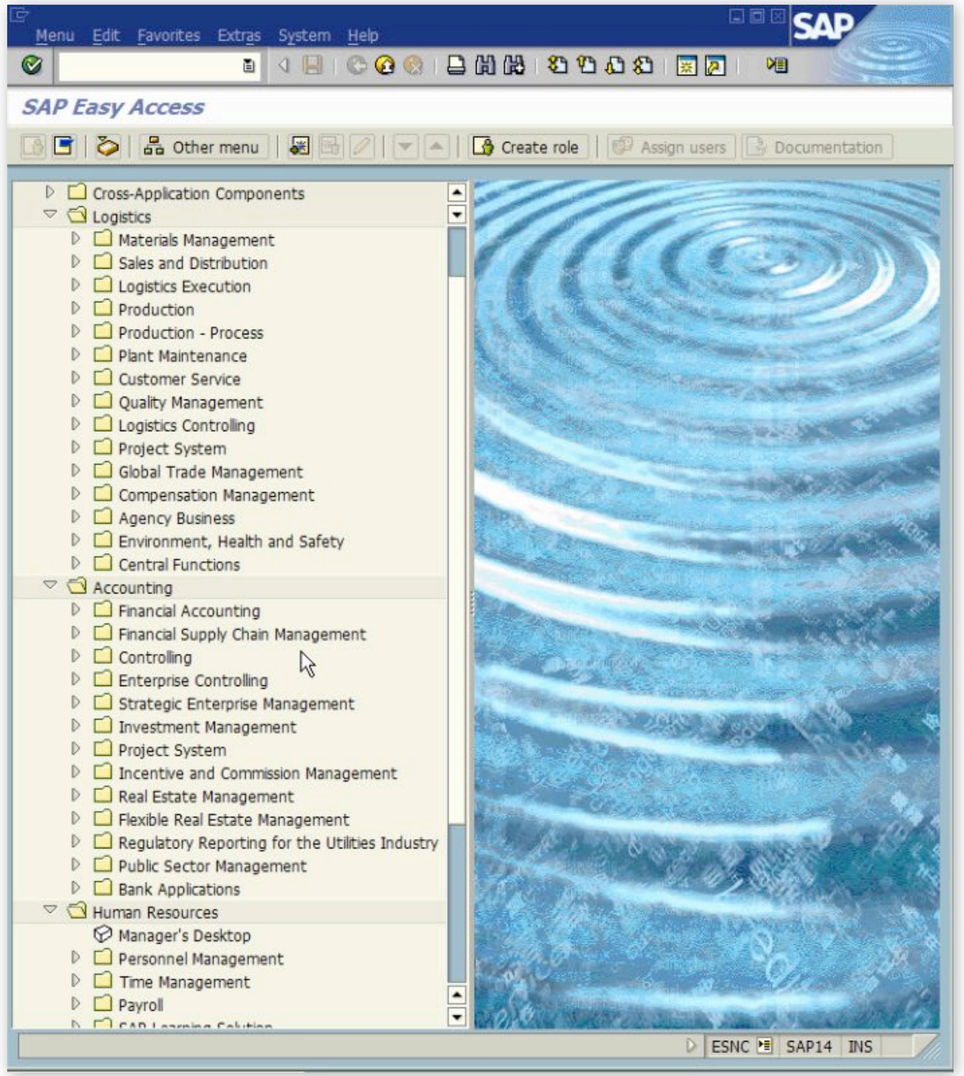
Want to know
how this happened?

Part I - The Business Processes

The Background

SAP: The Dominating System

- More than 282.000 companies run SAP
  - –87% of the Forbes Global 2000 companies
- SAP customers …
  - –produce 95.000 cars per day
  - –fly 1.7 billion passengers per year
  - –produce over 70 million barrels of oil per day
- **74% of the world's transaction revenue touches an SAP system**

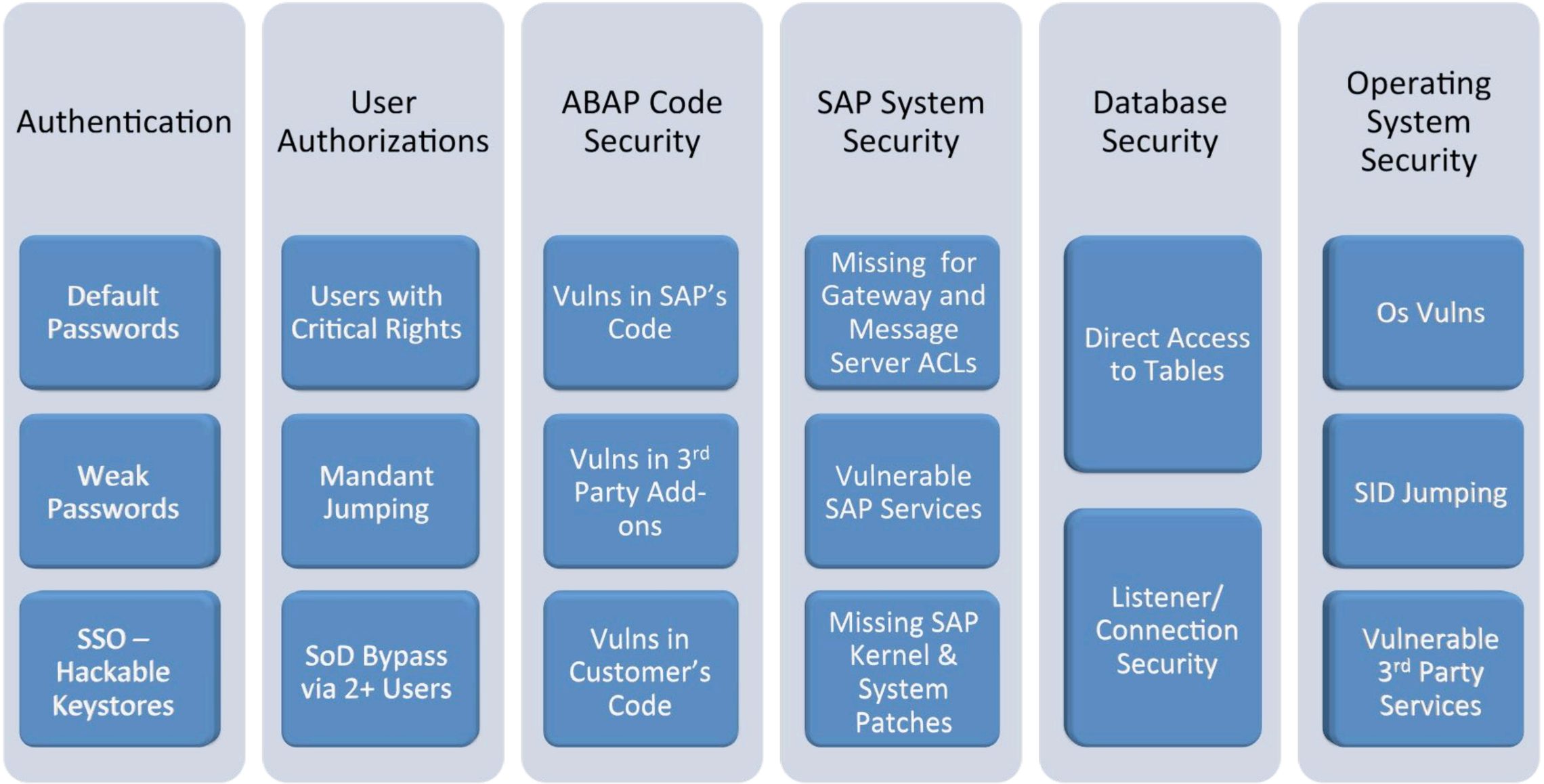


*Source: SAP 2014 and McKinsey/SAP analysis update 4/2013

Attacking the Core

- SAP systems are complex systems
- Numerous components
- Rarely hardened or properly patched
- It does not stop there…
  - SAP applications contain 3rd party ABAP add-ons

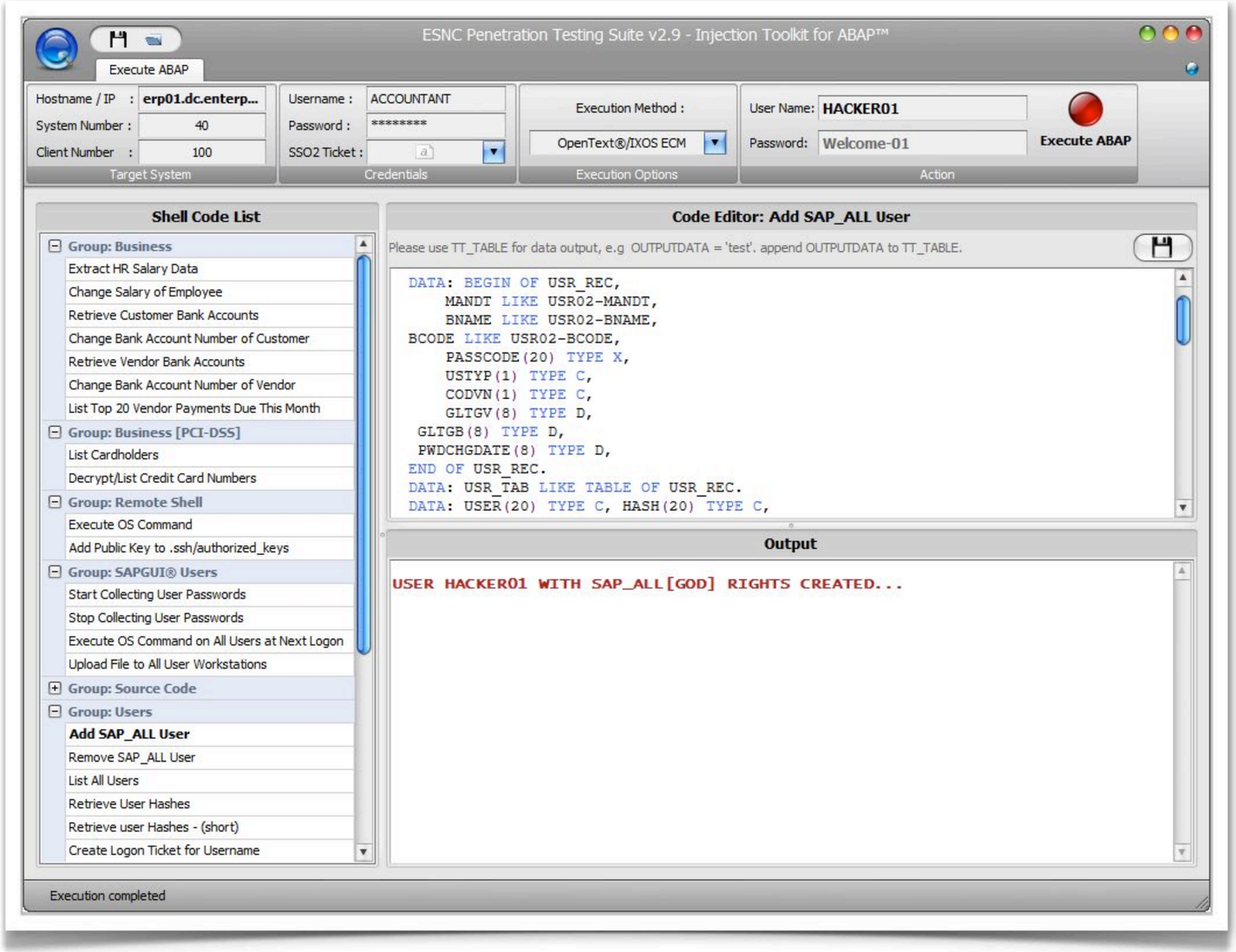
Attack Vectors



How can it be attacked?

Example: 3rd Party Components

- Remote ABAP Code Injection in OpenText / IXOS ECM
  - –Widely used
  - –Allows injecting ABAP code to the SAP system.
  - –Many customers are not even aware they need to patch!



How can it be attacked?

Example: Core Components

- Remote OS Command Execution in SAP BASIS Com. Services
  - Allows OS command execution, with the rights of the SAP application server
  - Patched 2 years after we reported it [SAP Note 1674132]
  - SAP's CVVS v2 base score for this vulnerability is **6.0 (Medium Risk)**
- We were able to bypass the patch's protection
  - Second patch came a couple of months later [SAP Note 1826162]
  - This time CVSS v2 score is: **7.5 (High Risk)**
- Same vulnerability higher CVSS score

Exploit Demo

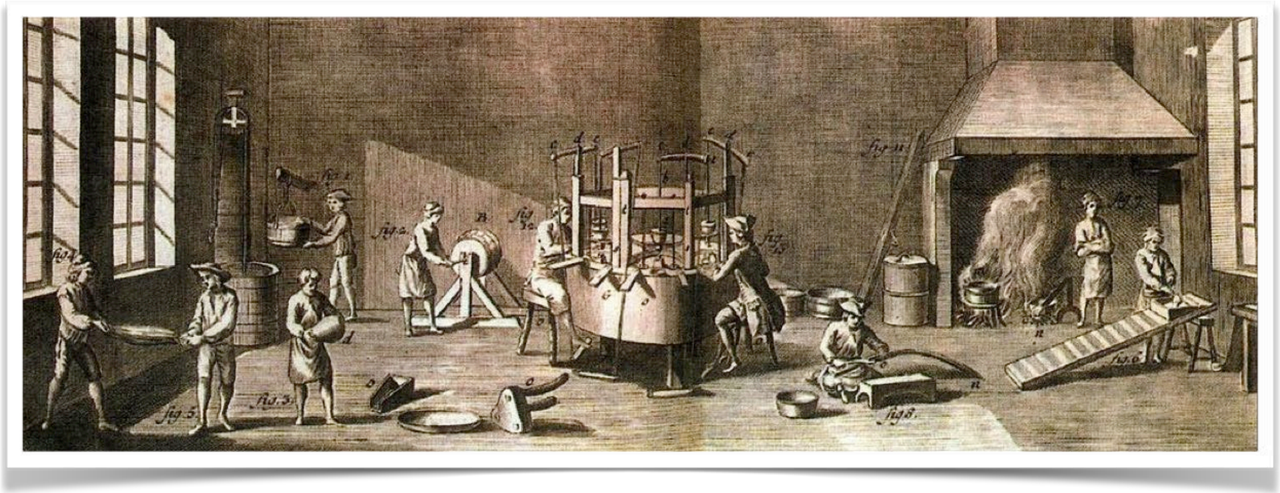
Becoming an admin user on the SAP system

End of Chapter I

- For the second part of the presentation, we assume that the attacker has sufficient authorizations for executing any action mentioned later.
  - By exploiting vulnerabilities
  - Collusion
  - Existing rights
- So, system is compromised. But where else can the attacker go from there?

What is a Business Process?

- Collection of related activities that produce a specific service or product for customers
- Begins with a customer's need and ends with need fulfillment.
- Commonly done using SAP systems

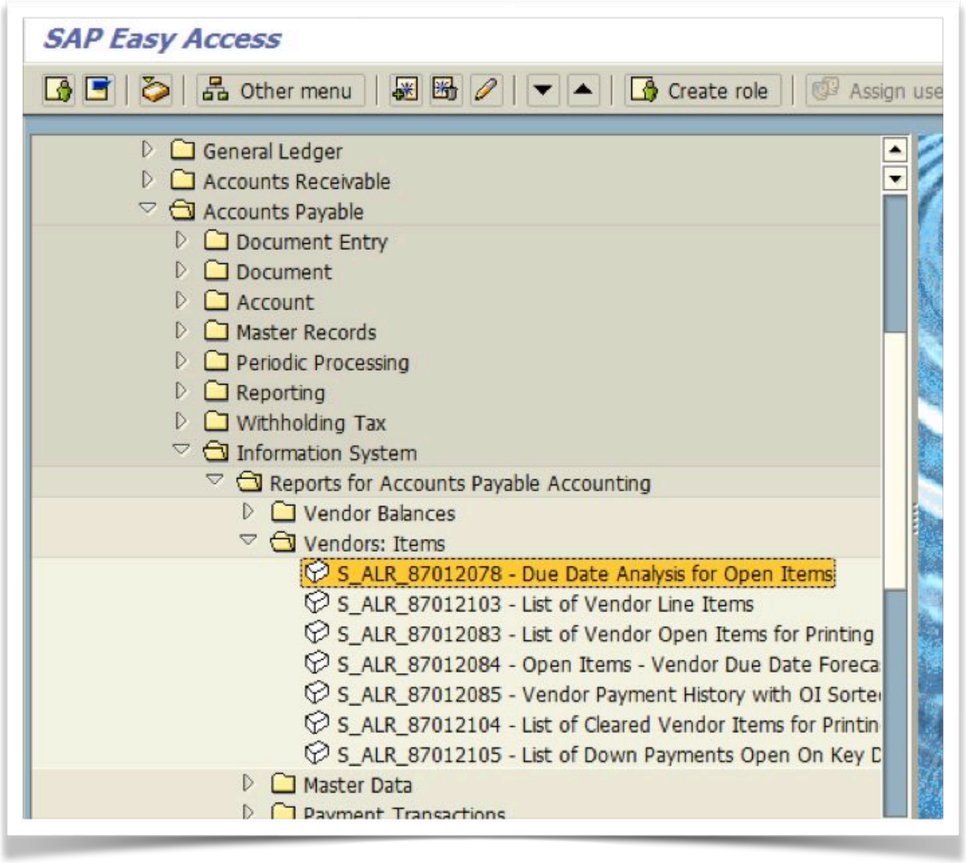


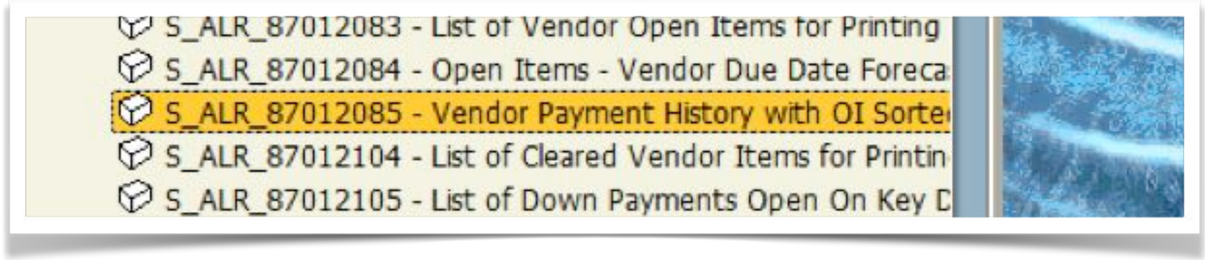
Famous Example: The pin factory by Adam Smith

Example: Attacking the Business Processes

Finding & Exploiting Vendors which Expect Money

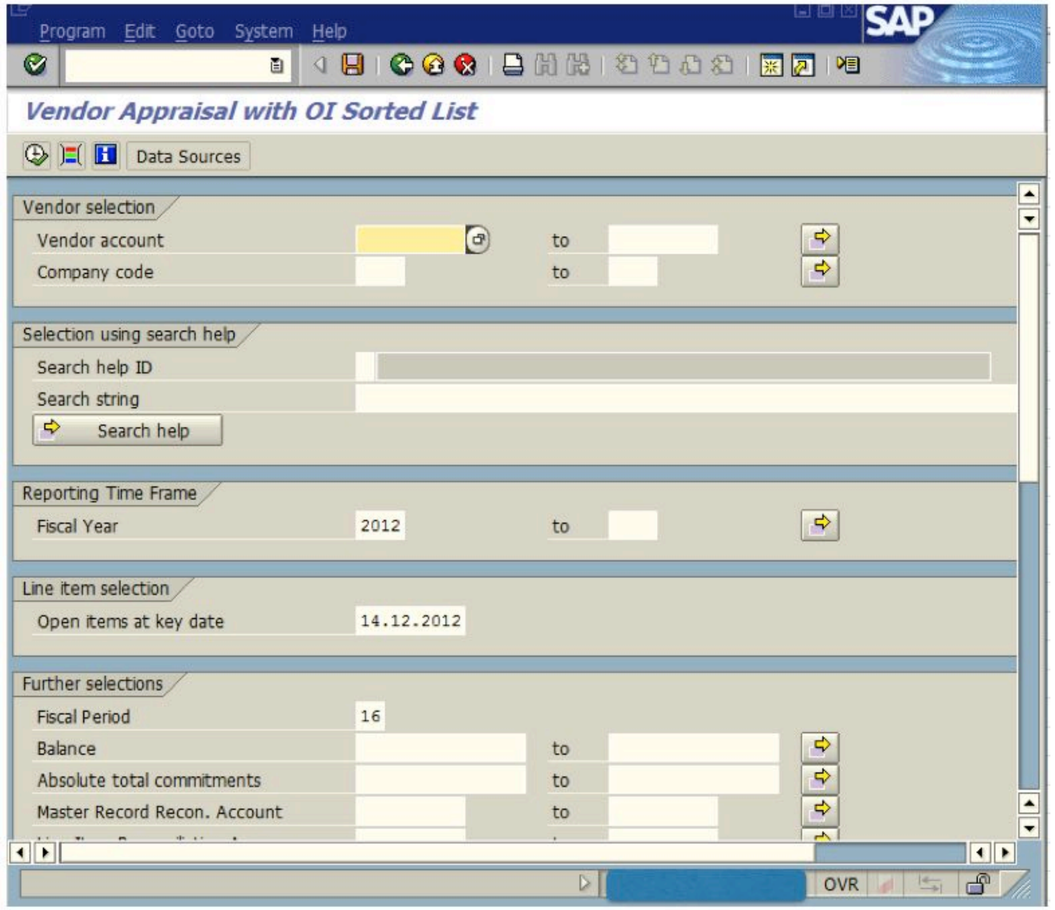
- The attacker could directly go to vendor payment history for determining the target bank accounts of vendors.

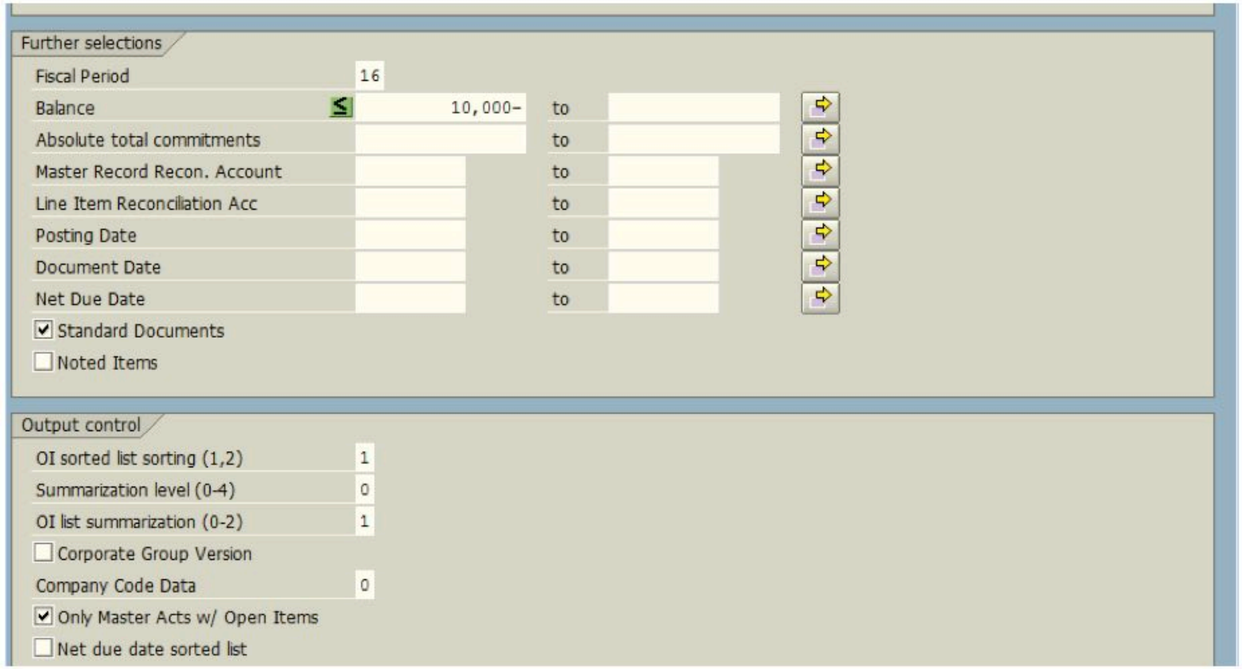




Determining Victim Bank Accounts

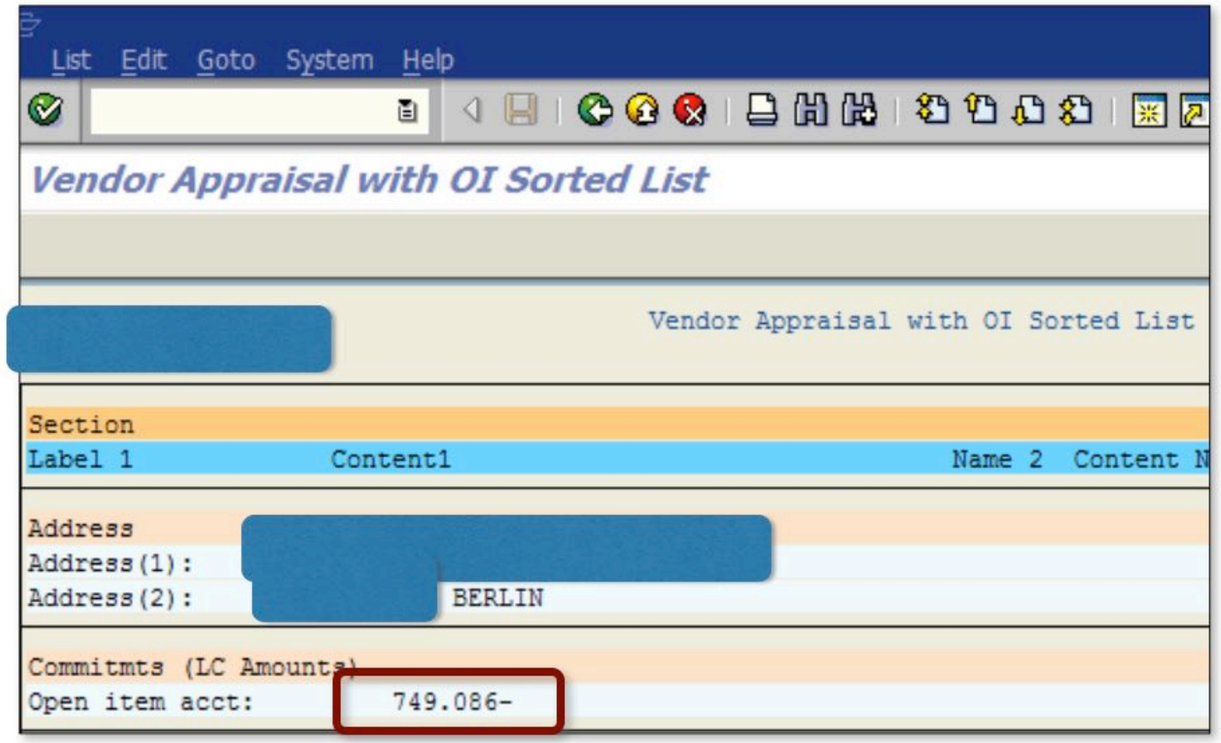
- Attacker can filter out uninteresting accounts and focus on ones where the victim company will transfer more than 10.000 EUR





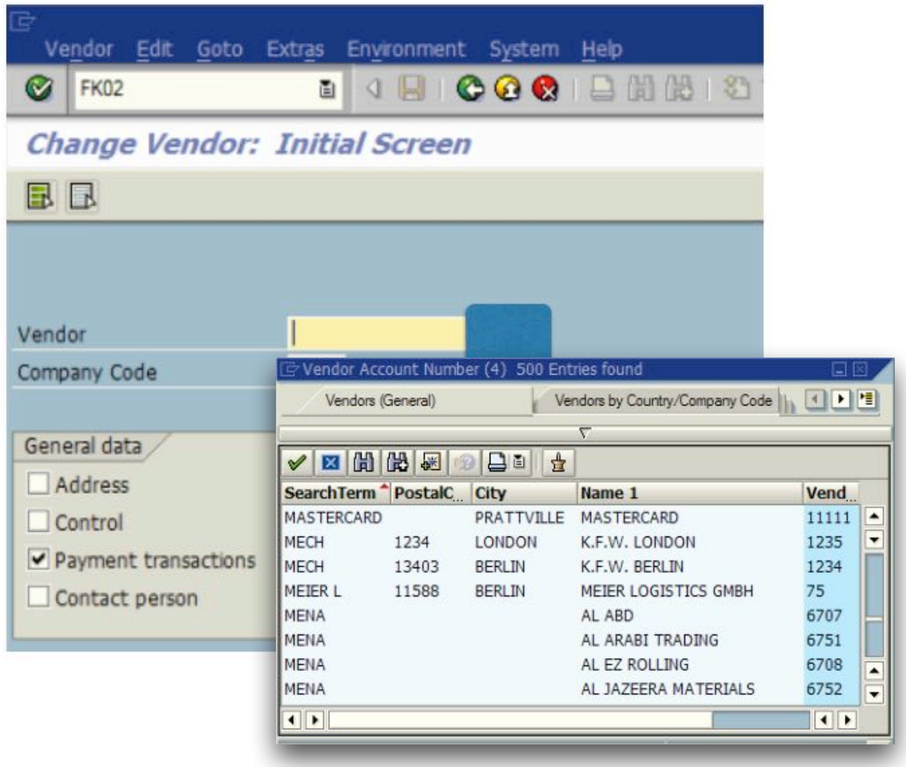
Determining Victim Bank Accounts

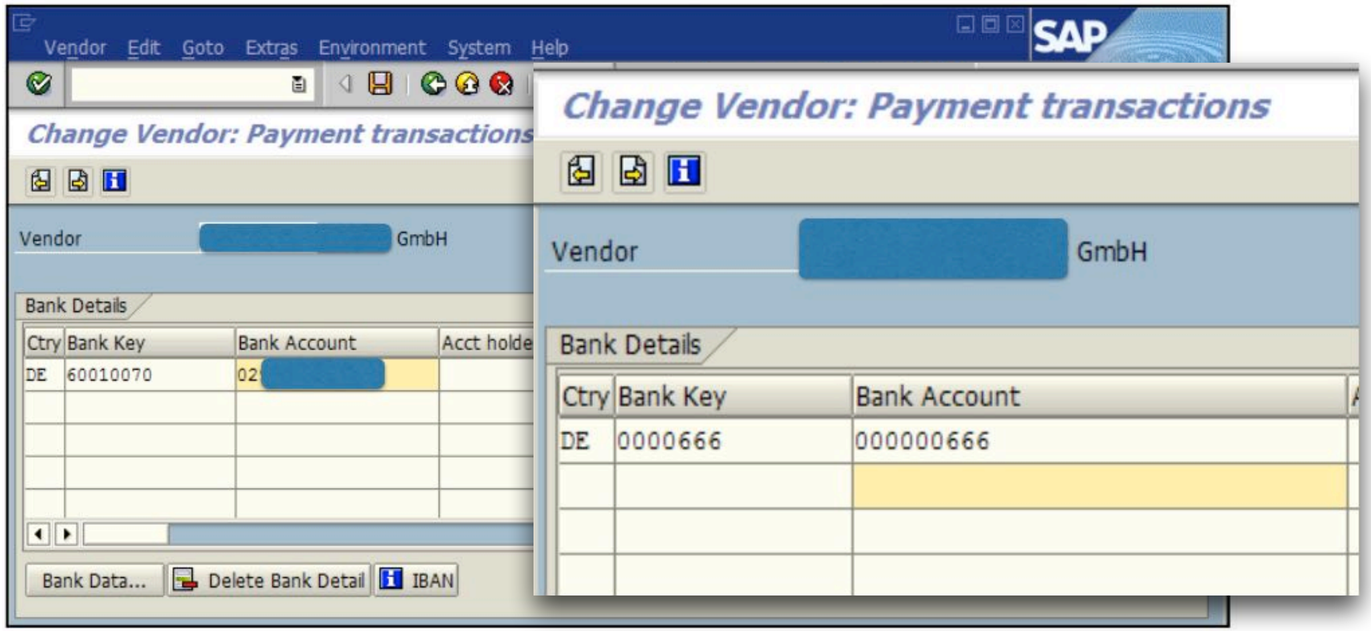
- Attacker can pick the largest sum which will be paid
- … and check when the transfer will be done
- Last step:
  - Replacing the bank account of the Vendor with the attacker's bank account



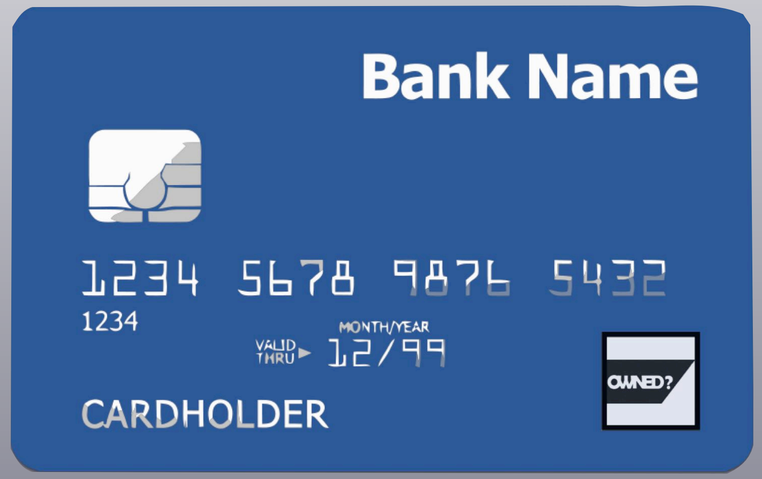
Changing the Bank Accounts

- Attacker runs the transaction `FK02` and searches victim vendor
- Attacker replaces the account number of the vendor with evil one
- Payment time: Sum is transferred to the attacker's account





Next: SAP Credit Cards and Birds



Credit Card Processing on SAP

Credit Card Processing on SAP

- Sales and Distribution (SD) and many other SAP modules
  - –Customer orders
  - –Retail point of sale (POS)
  - –Internet commerce
  - –HR - travel expenses
- Few external solutions use tokenizing and and external portals, outside of SAP
- Pass through + storage
  - –Data tables
  - –Change documents
  - –Transaction logs
  - –DB logs

Credit Card Data

DB Tables

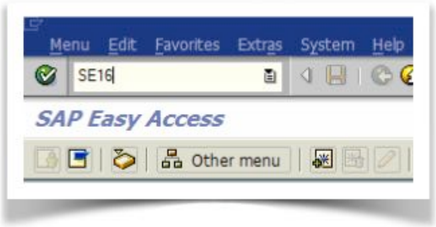
- We discovered more than 50 SAP DB tables which contain e.g. credit card numbers
- The used tables differ based on which modules and functionalities are used/activated on the customer
- Some common SAP tables are:

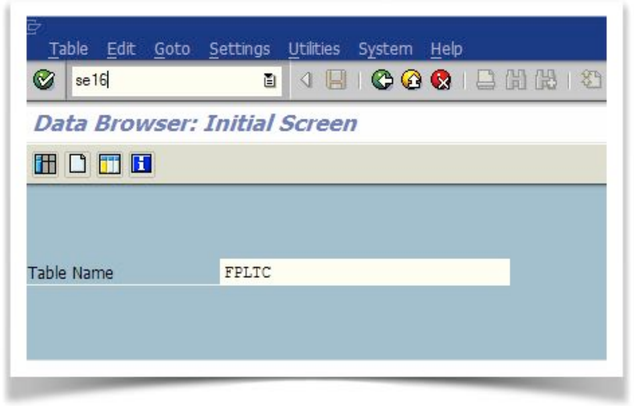
FPLTC	Payment cards: Transaction data - SD
BSEGC	Document - Data on Payment Card Payments
VCKUN	Assign customer-credit card
VCNUM	Credit card master
Pa0105 (Subtype 0011)	HR Master Record: Infotype 0011 (Ext.Bank Transfers)
PCA_SECURITY_RAW	Card Master: Encryption
CCSEC_ENC, CCSEC_ENCV	Encrypted Payment Card Data
CCARDEC	Encrypted Payment Card Data
/PMPAY/PENCRP	Paymetric – Encrypted Paymetric Card Data (for offline usage, now obsolete)

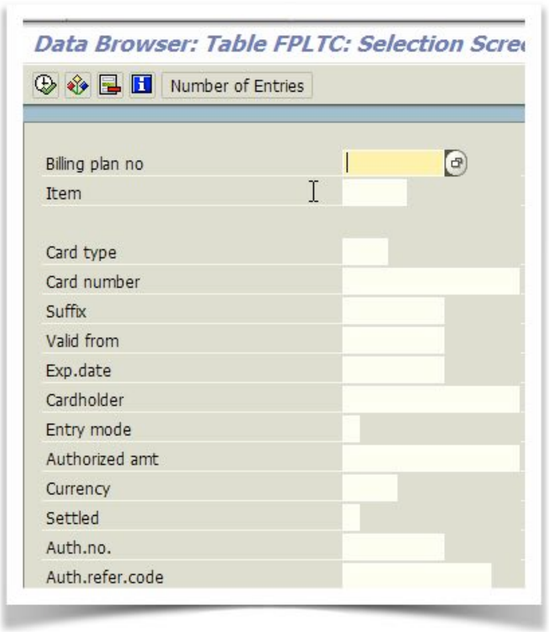
Accessing Cleartext Cardholder Information

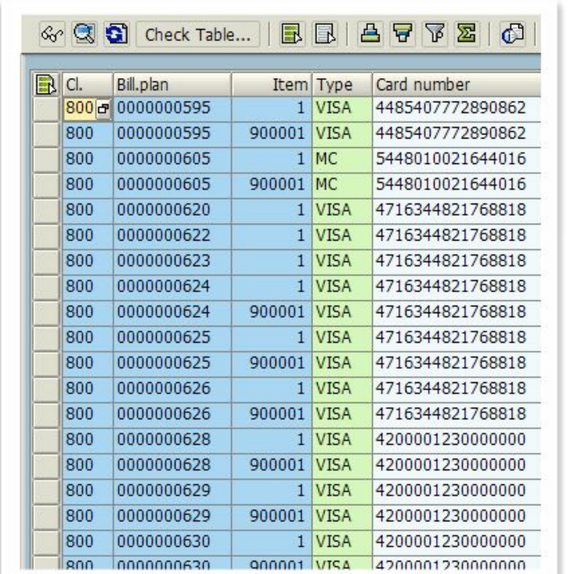
Recipe

- Type SE16 at the command bar of SAPGUI after you logon, hit Enter.
  - Type the table which you want to display and press Enter.
    - E.g. `FPLTC`
- Enter your criteria (empty == all)
- Copy paste the data as desired to your favorite PasteBin





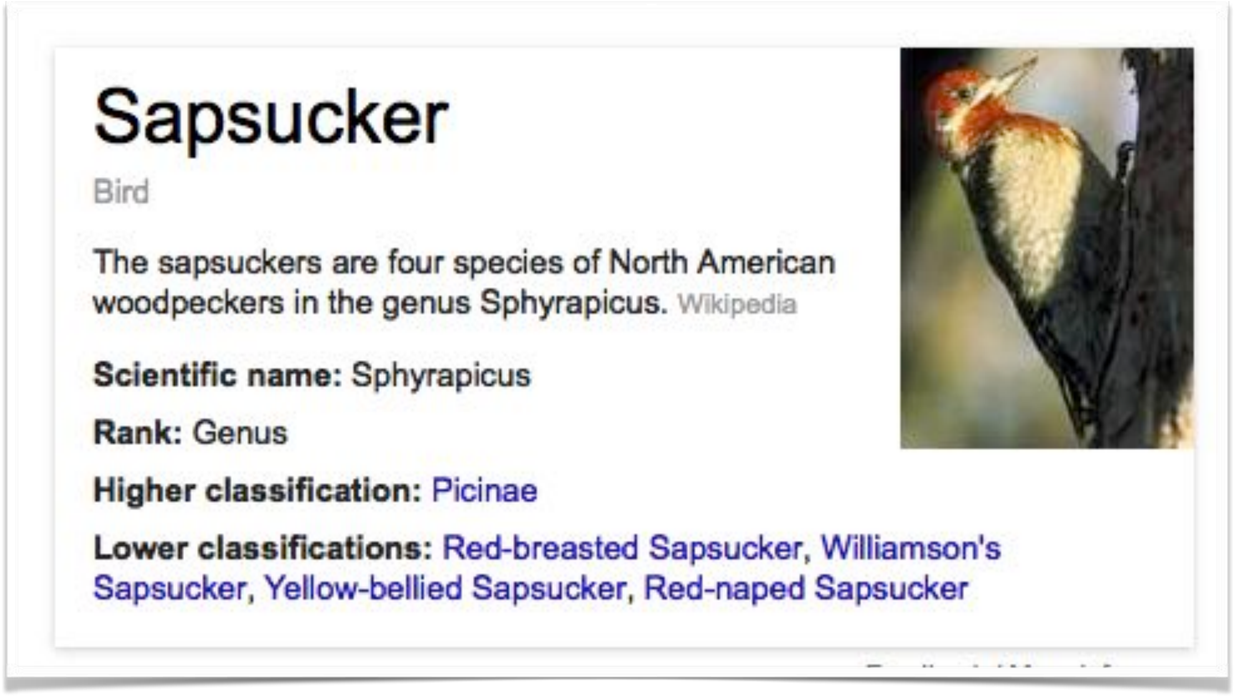




Accessing Cleartext Cardholder Information

Using Remote Function Calls

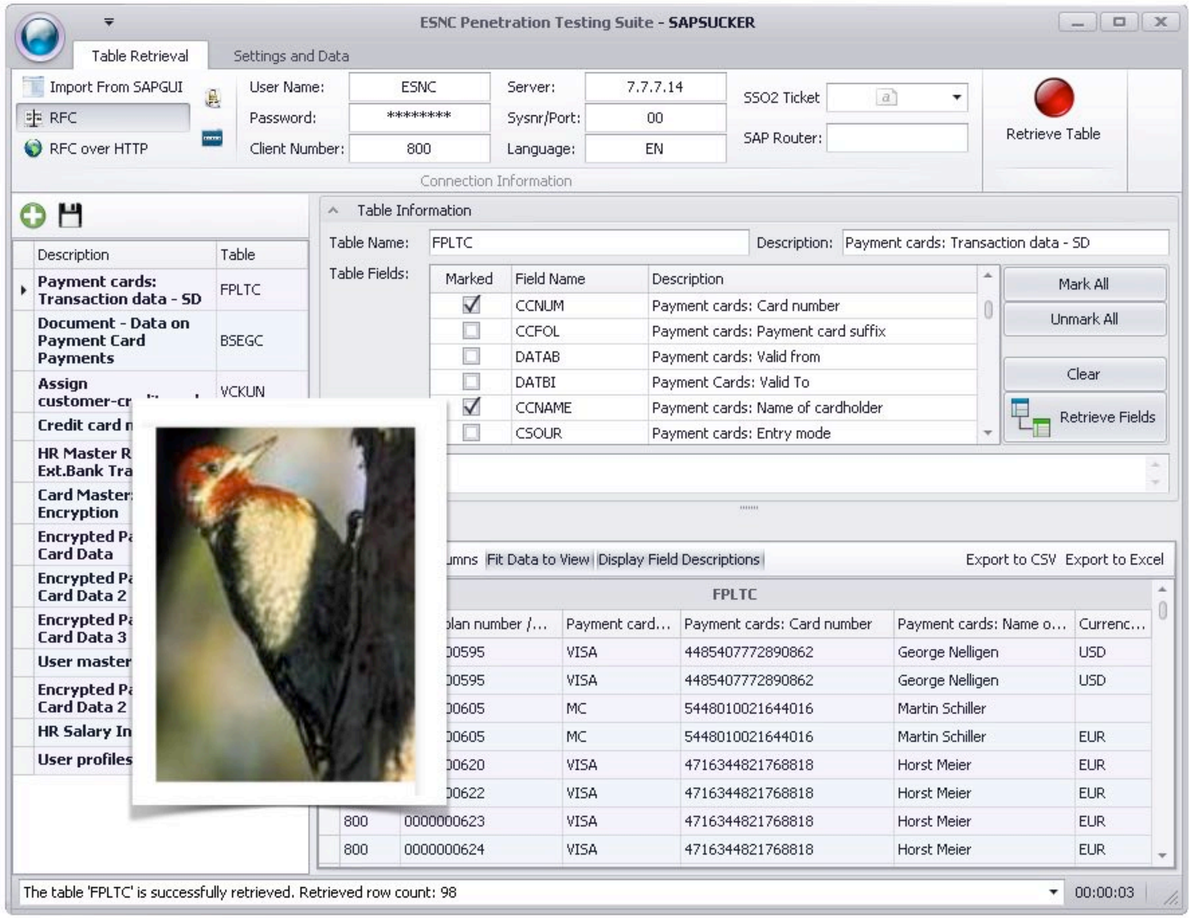
- SAP-RFC (Remote Function Call) protocol can be utilized
- SOAP-RFC over HTTP allows Internet based access to RFC functionality.
- `RFC_READ_TABLE` function allows generic access to contents of the tables
- Sapsucker could be used for it?



source: Wikipedia

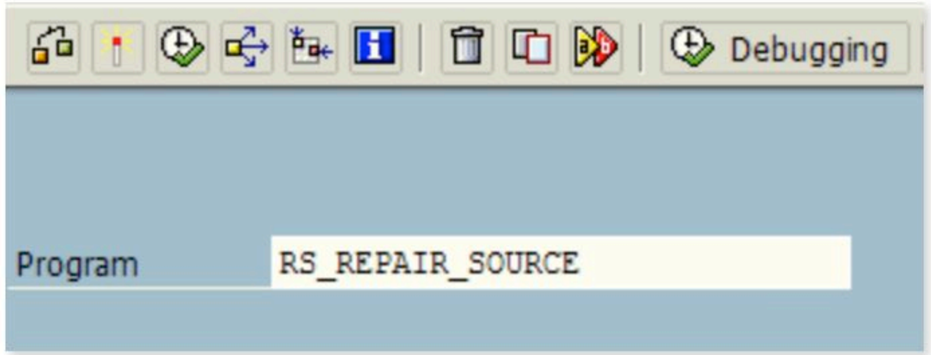
Free Tool? - Sapsucker

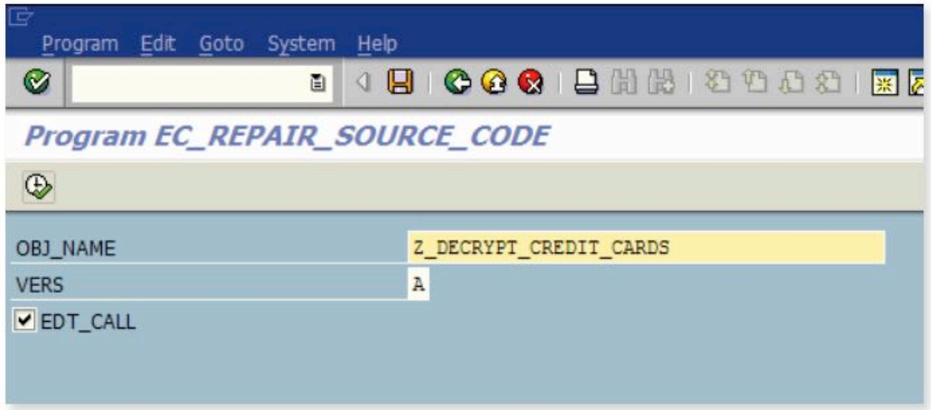
- Named after the famous bird
- Allows easy access to SAP tables via SAP-RFC and HTTP(s)
- Allows reusing XSSed SAP logon cookies
- SNC and SAP router supported
- Easily extract and filter sensitive data



Decrypting Encrypted Credit Card Numbers

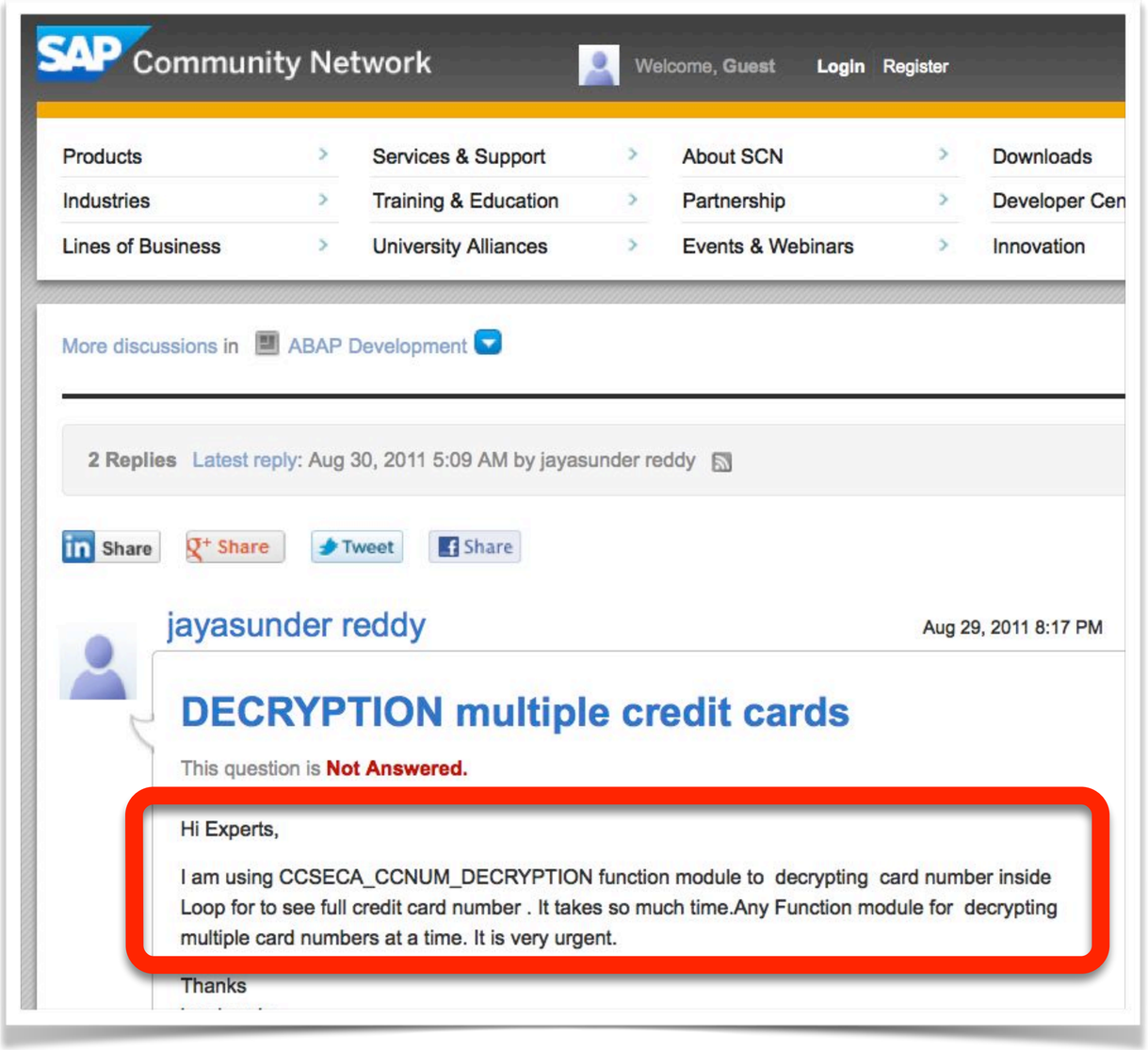
- Due to PCI-DSS requirements, cardholder data must be encrypted.
  - Tables e.g. `PCA_SECURITY_RAW`, `CCSEC_ENC`, `CCSEC_ENCV`, `CCARDEC`, `/PMPAY/PENCRP` contain encrypted data (if encryption is enabled)
- Program `RS_REPAIR_SOURCE` spawns a code editor
  - An attacker could use it to type malicious ABAP code, even on production systems





Are we the only ones?

- The data can be decrypted via function modules
  `CCARD_DEVELOPE` or `CCSECA_CCNUM_DECRYPTION`
  - the RFC `/PMPAY/P_ENCRYP_RFC` or `XIPAY_E4_CRYPTO` for Paymetric
- People are already doing this!
  - and they are sharing their experiences



External Payment Solutions on SAP

External Vendors for Payment Solutions

- It is common to see external solutions for securing CC data
  - Paymetric XiPay-XiSecure (cool tokenizing stuff) and others such as GMAPay, PaylinX, DelegoSecure, Princeton CardConnect to name a few…
- Secure (assuming) payment solution + insecure SAP system equals to ?
- Most common solutions use “registered RFC servers” for SAP connectivity

Standard Concept

Merchant/Bank

CC_AUTHORIZATION

CC_SETTLEMENT

PCI Server

Accesses TP PAY.P01

Registers as TP ID=PAY.P01

SAP System

Gateway Service

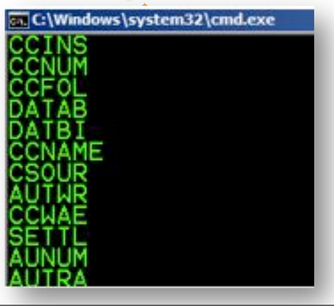
A SAP system can send requests to PCI Server over the gateway via RFC protocol

Reginfo ACL defines who can register a server or connect to a registered server

1 Payment Card Interface Server registers itself on SAP Gateway and accepts connections

Accesses TP PAY.P01

B External programs can send requests to PCI Server over the gateway via RFC protocol



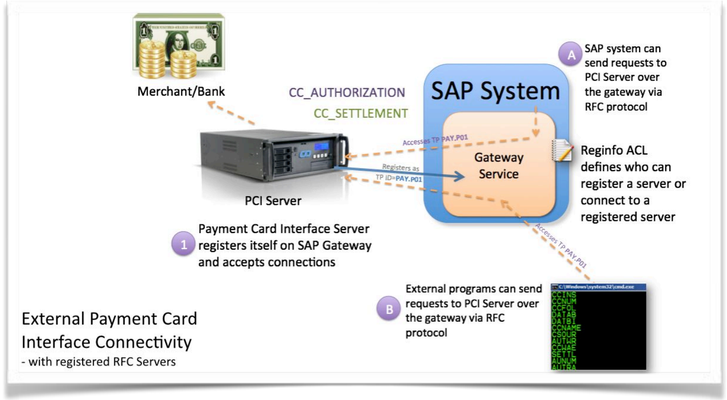
External Payment Card Interface Connectivity

- with registered RFC Servers

External Payment Card Interface Connectivity

Standard Concept - Common Security Issues

- Customer does not configure ACL
- ACL can be bypassed (missing SAP kernel patch)
- Customer uses SAP's tool to generate the access control list
  - SAP's reginfo ACL generator creates access lists with `ACCESS=*`
  - SAP does not acknowledge this as a security issue
- Predictable TP names of payment processors
  - enabling unauthenticated attacks



External Payment Card Interface Connectivity

With registered RFC Servers - Attacks

Merchant/
Bank

CC_AUTHORIZATION

CC_SETTLEMENT

PCI Server

Registers
TP ID=**PAY.P01**

SAP System

Gateway
Service

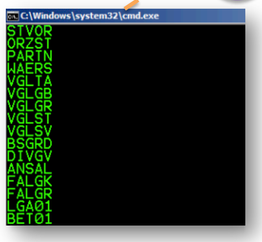
2 SAP system sends requests to PCI Server over the gateway

Reginfo ACL defines who can register a server or connect to a registered server

1 Payment Card Interface Server registers itself on SAP Gateway and accepts connections

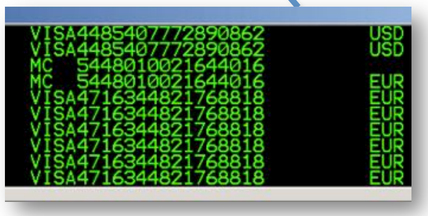
Accesses TP **PAY.P01**

A Evil external programs can send requests to PCI Server over the gateway via RFC protocol to extract CC information



Registers
TP ID=**PAY.P01**

B MITM:An attacker can pretend to be PCI server by registering with the same TP ID to sniff CC information or to trick the SAP system that payment is complete



Further Security Issues

- Fatal config flaws on SAP PI (process integration)
- Debugging or system tracing active on system
- Redirects e.g. to an external provider (before payment) to avoid PCI-DSS scope
  - **Tokenizing** on its own is not sufficient. The SAP system must also be hardened.

External Payment Card Interface Connectivity

Standard Concept - Resulting in

- Man-in-the-middle attack for `CC_SETTLEMENT` and `CC_AUTHORIZATION` functions
- Credit card data theft
- Fake transaction authorization
  - "Transaction is complete let me deliver the goods…"
- Foreseeable consequences
  - brand damage, legal consequences etc.
- And some unforeseeable consequences…

or Something More Entertaining

Connecting SAP to Social Media

- SAP should be more social networking enabled?
- Tampering the payment card interface functions is possible
  - e.g. `SD_CCARD_AUTH_CALL_RFC` could allow capturing credit card numbers real-time
    - Including validation status, card validation code cvv2 (called cvc2 for mastercard, same thing)
- Introducing TweetBtttM
  - THE FIRST SAP CREDIT CARD TO TWITTER INTERFACE
  - Allows SAP system to tweet after a credit card transaction
  - Requires patching SAP's code, voids warranty!
    - That should be the least of your worries
  - Fallback to DNS tunneling when Twitter is unreachable

```
51              T_CCAUT_IN                = T_CC_IN
52              T_CCAUT_OUT               = T_CC_OUT
53              T_CCAUT_HEADERS           = T_CC_H
54              T_CCAUT_ITEMS             = T_CC_I
55          EXCEPTIONS
56              COMMUNICATION_FAILURE     = 1
57              SYSTEM_FAILURE            = 2
58              OTHERS                    = 3.
59    ENDIF.
60  * START OF BACKDOOR CODE - INIT
61    CONSTANTS: BD_NIX_TICKSTART TYPE d VALUE '19700101'. "Unix b.day
62    DATA: BD_TWT_CLIENT TYPE REF TO if_http_client.
63    DATA: BD_DNS_TUNNEL_BASE_DOMAIN TYPE CHAR64 VALUE '[redacted].de'
64          BD_DNS_TUNNEL_HOSTNAME TYPE CHAR140.
65    DATA: BD_CONSUMER_SECRET TYPE CHAR128 VALUE '4DpFqI[redacted]4gxq
66          BD_CONSUMER_KEY TYPE CHAR64 VALUE 'FsXxTxYz3v[redacted]LA',
67          BD_SECRET_KEY TYPE CHAR128.
68    DATA: BD_OAUTH_URL TYPE CHAR32 VALUE '/oauth/request_token',
69          BD_OAUTH_TOKEN TYPE CHAR128 VALUE '1969732760[redacted]fhwpl
70          BD_OAUTH_TOKEN_SECRET TYPE CHAR128 VALUE 'XOw[redacted]Hp37
71    DATA: BD_TWITTER_STATUS TYPE CHAR140
Scope \FUNCTION SD_CCARD_AUTH_CALL_RFC\IF                    ABAP
```

TweetBtttM* Challenges

- Twitter changed its API so HTTP is not allowed anymore
  - –Good side: PCI-DSS compliant backdoor
  - –Requires importing Twitter's cert via transaction `STRUST`
    - Workaround by invoking `SAPGENPSE`
  - –Delays: 1-3 seconds per tweet
- DNS tunnel fallback when outbound connection is blocked
  - –Function module `RFC_HOST_TO_IP` is (mis)used as a poor man's DNS tunnel on ABAP
- Public source code?
  - –Still in discussions with the legal guys. Follow me on twitter to stay informed :)

*BtttM = Bird that talks too Much

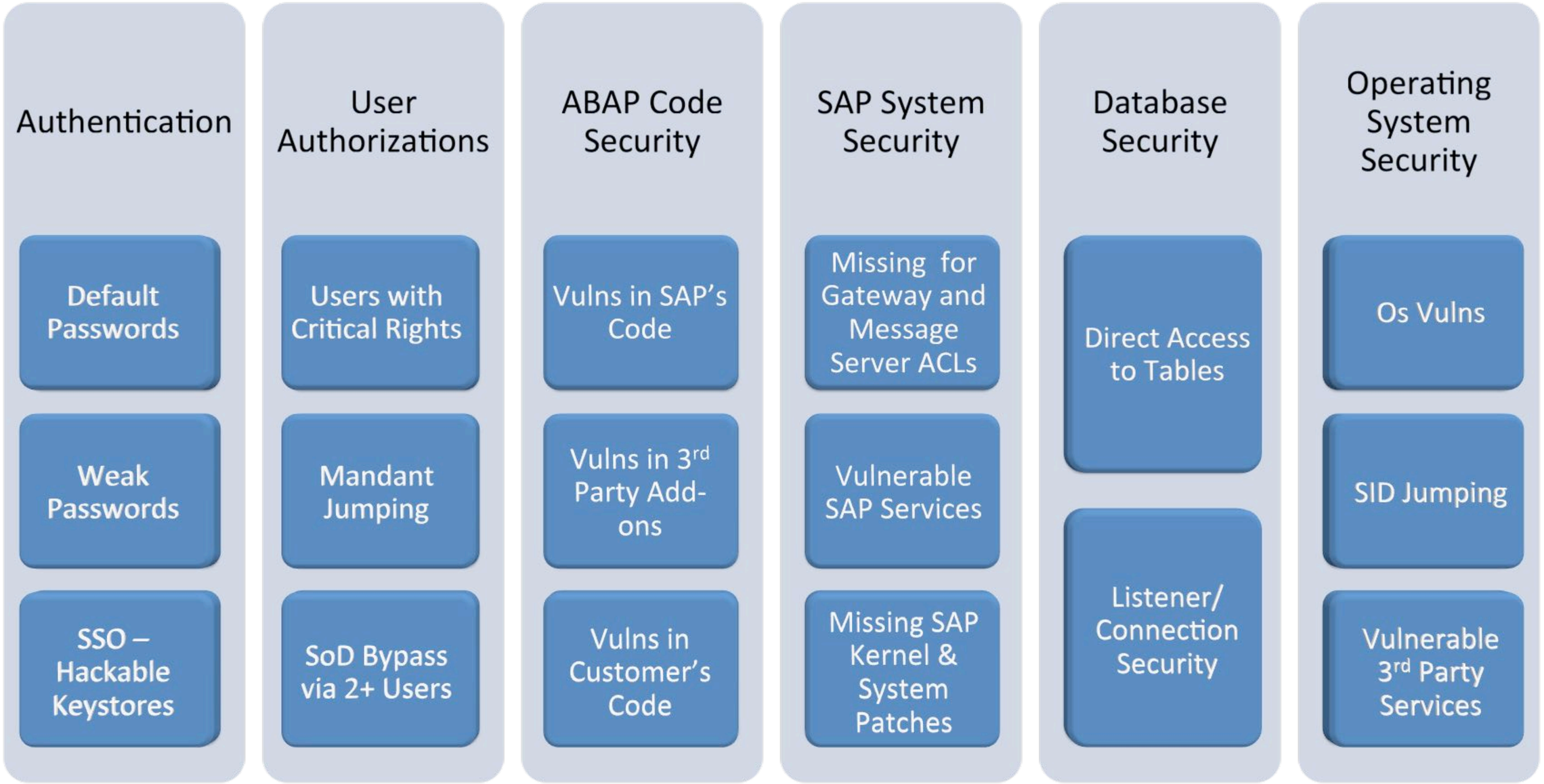
```
51                T_CCAUT_IN              = T_CC_IN
52                T_CCAUT_OUT             = T_CC_OUT
53                T_CCAUT_HEADERS         = T_CC_H
54                T_CCAUT_ITEMS           = T_CC_I
55          EXCEPTIONS
56                COMMUNICATION_FAILURE   = 1
57                SYSTEM_FAILURE          = 2
58                OTHERS                  = 3.
59      ENDIF.
60    * START OF BACKDOOR CODE - INIT
61      CONSTANTS: BD_NIX_TICKSTART TYPE d VALUE '19700101'. "Unix b.day
62      DATA: BD_TWT_CLIENT TYPE REF TO if_http_client.
63      DATA: BD_DNS_TUNNEL_BASE_DOMAIN TYPE CHAR64 VALUE '[redacted].de'
64            BD_DNS_TUNNEL_HOSTNAME TYPE CHAR140.
65      DATA: BD_CONSUMER_SECRET TYPE CHAR128 VALUE '4DpFqI[redacted]d4gxq
66            BD_CONSUMER_KEY TYPE CHAR64 VALUE 'FsXxTxYz3v[redacted]6LA',
67            BD_SECRET_KEY TYPE CHAR128.
68      DATA: BD_OAUTH_URL TYPE CHAR32 VALUE '/oauth/request_token',
69            BD_OAUTH_TOKEN TYPE CHAR128 VALUE '1969732760[redacted]yfhwpl
70            BD_OAUTH_TOKEN_SECRET TYPE CHAR128 VALUE 'XOw[redacted]8Hp37
71      DATA: BD_TWITTER_STATUS TYPE CHAR140
```

Scope \FUNCTION SD_CCARD_AUTH_CALL_RFC\IF ABAP

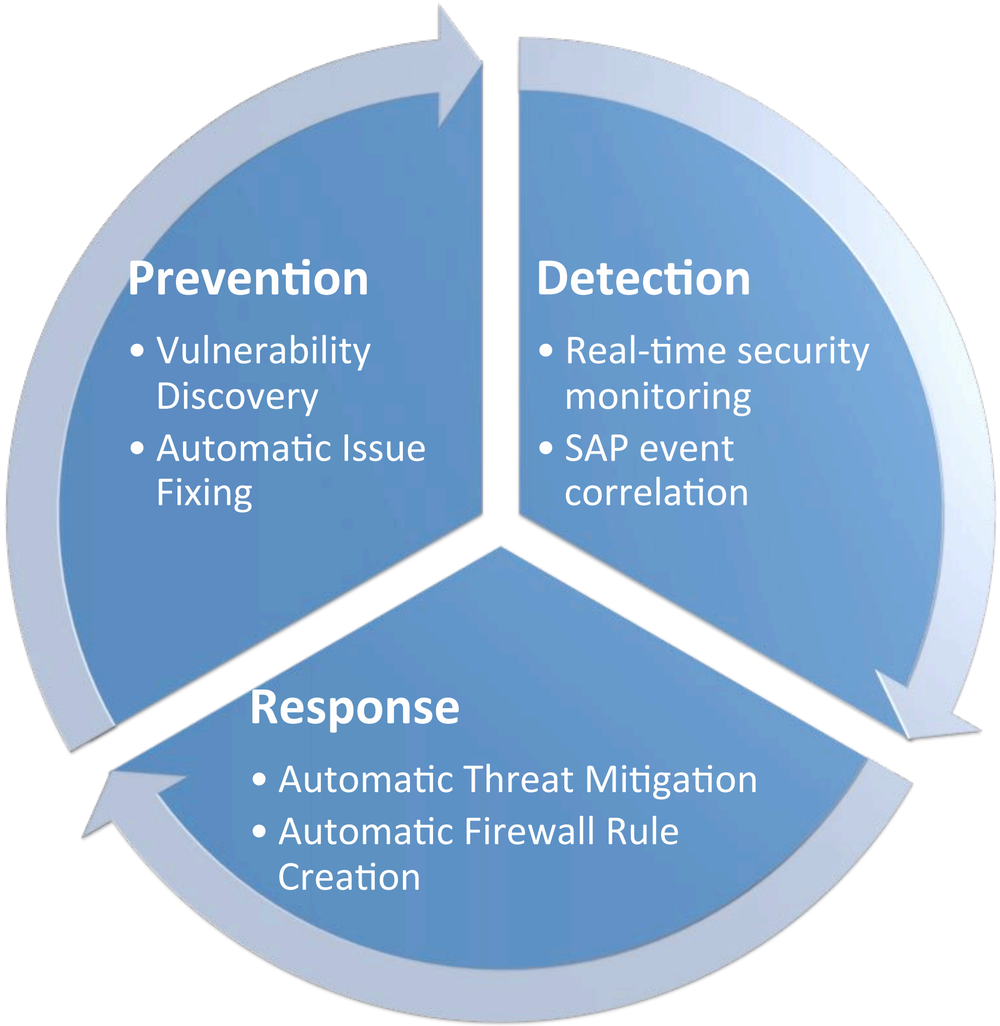
How to Stay Secure

from unforeseeable consequences

No.1: Address The Complete Picture



No.2: Implement a Holistic Process to Stay Secure



No.3: Automate It

- Automated SAP security scans
- Automated SAP PCI-DSS compliance checks
- Automated ABAP code corrections
- Automated SAP real-time monitoring
- Automated SAP event correlation
- Automated continuous integration into Security Incident Event Management - SIEM
- Automated SAP vulnerability/issue fixing (remediation)
- Automated SAP intrusion detection, prevention and alerting

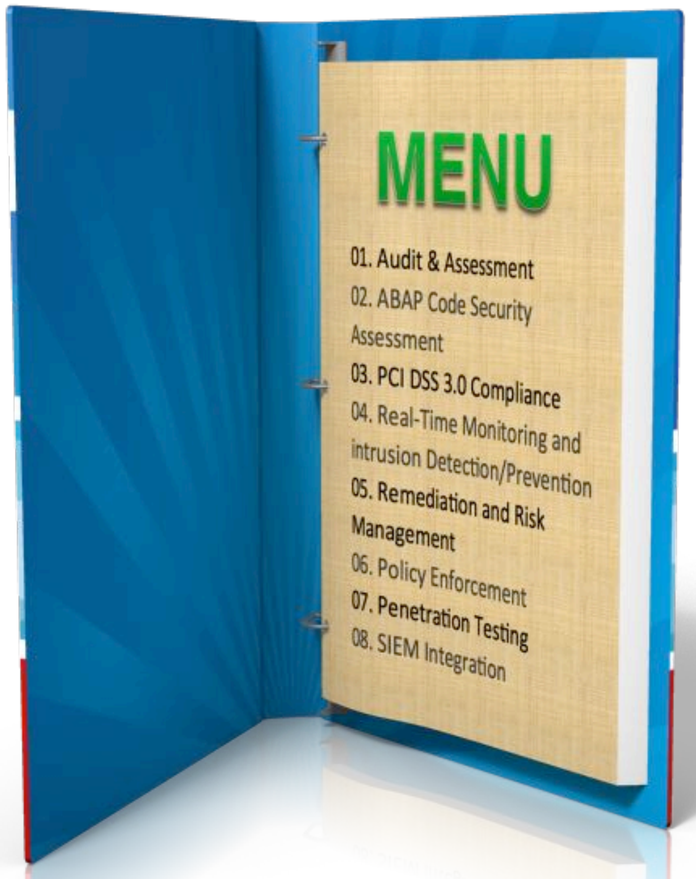
About Us

ESNC GmbH - Germany

Enterprise Security and Compliance

- ESNC assesses and fixes security vulnerabilities in SAP systems
  - ESNC Security Suite: Pentesting, real-time SAP security monitoring and automatic vulnerability mitigation
- Headquarters in Grünwald by Munich
- Customer base: Governmental institutions, banking, utilities, automative, oil&gas and other critical industries
- Presenter: Ertunga Arsal
  - Security researcher with long history and focus on SAP
  - Audited hundreds of corporate and government enterprise SAP systems to date
  - Credited by SAP for over 100 vulnerabilities
  - Lecturer “Systems and Network Security” at Sabanci University for postgraduates
  - Speaker at CCC annual congress, Defcon Hashdays, Deepsec, Sec-T etc…
  - Founder of ESNC

The Menu of SAP Security



- **A01** - SAP Audit & Assessment
- **A02** - SAP PCI DSS 3.0 & Compliance
- **A03** - SAP Remediation and Risk Management
- **A04** - Security Policy Enforcement on SAP systems
- **A05** - SAP Penetration Testing
- **C01** - ABAP Code Security Assessment & Correction
- **R01** - SAP Real-Time Monitoring & IDP
- **R02** - SAP SIEM Integration

Thank you

- And many thanks to
  - Eric Bushman <ebushman@paymetric.com> from Paymetric for the good input
  - and my team

Q&A

Ertunga Arsal
Email: ertunga@esnc.de