Berner
Fachhochschule

BTI7301 Project 1

Mail Server Set-Up & Security-Hardening Script

Bern University of Applied Sciences

Frido Zurlinden
Ismael Riedo
Jan Henzi
Tutor: Dr. Simon Kramer

Contents

Introduction

Goals and Requirements

Technical Solution / Realization

Challenges

Berner Fachhochschule

Introduction

IT security for everyone

- Transparency
- Simple
- Vendor independent

Goals and Requirements

- Easy as pie “Complexity is the enemy of security”
- Brick by brick
- Power to the people (No vendor lock-in)
- Security by default

Technical Solution

Introduction

- Setup
- Firewall
- DNS
- SSH
- Mail

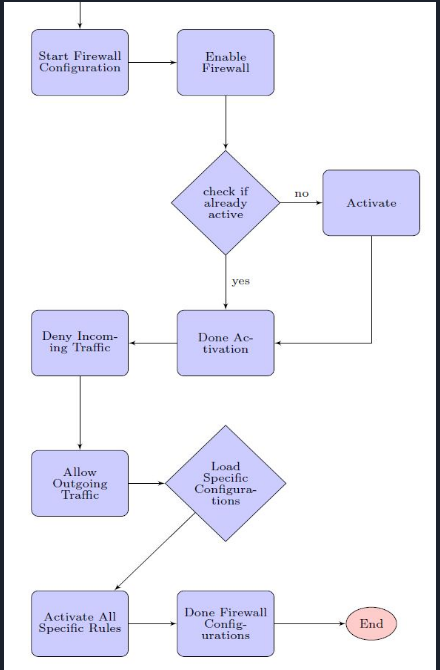
Setup

“Main Script” / “Entry point”

Flags - Leading pleasantly through the script

Handle State - What happens at the second time?

Firewall



src/fw/fw.sh

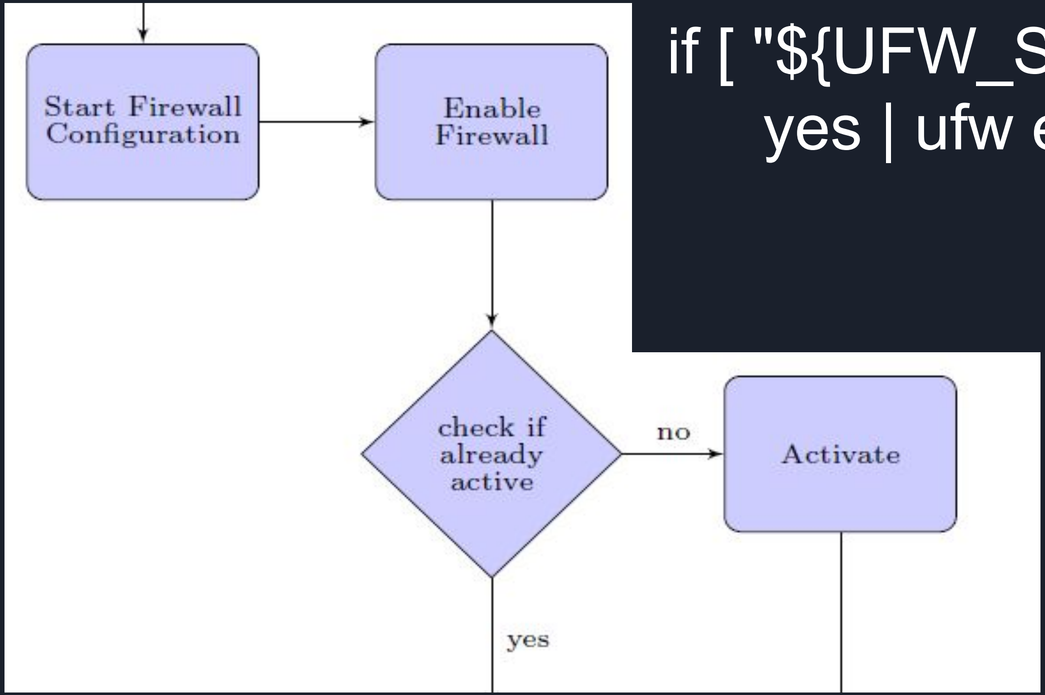
src/fw/enableUfw.sh

src/fw/controllTraffic.sh

src/fw/specificConfigurations.sh

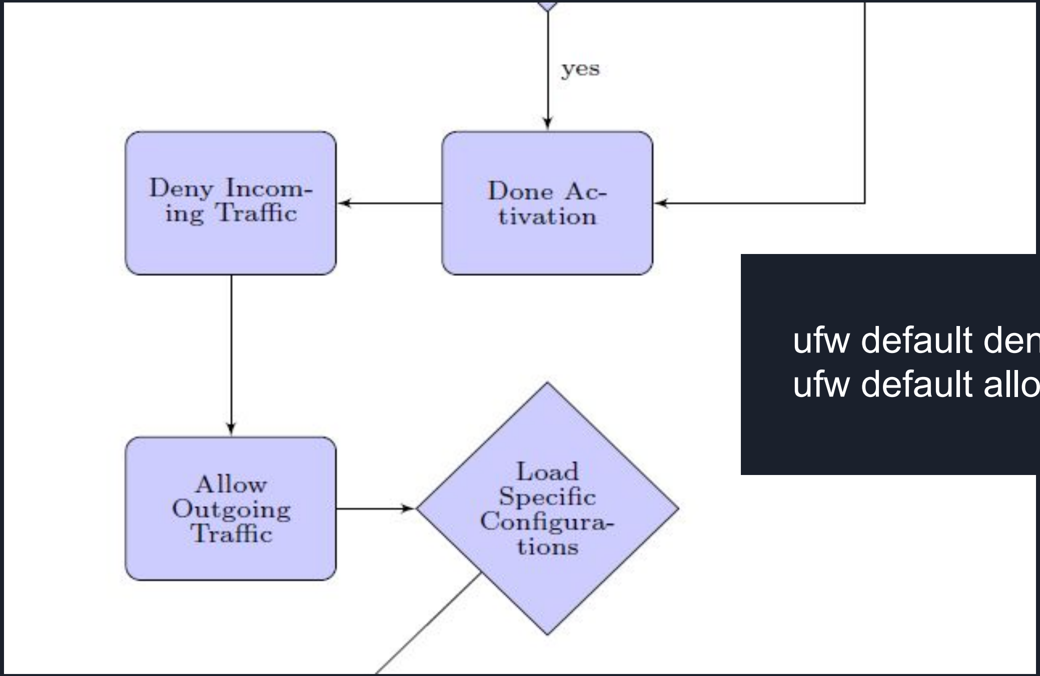
src/files/fw.conf

Firewall



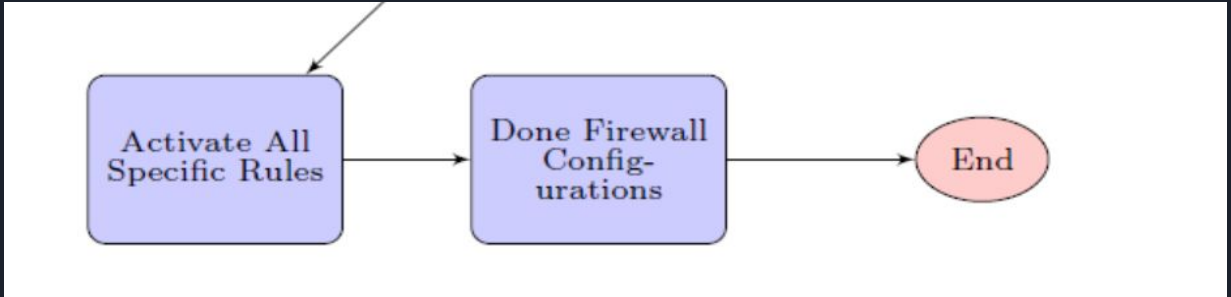
```
if [ "${UFW_STATUS}" == "inactive" ]; then
     yes | ufw enable > /dev/null 2>&1
```

Firewall



```
ufw default deny incoming
ufw default allow outgoing
```

Firewall



```
$ cat ../files/fw.conf
# SSH
allow - tcp - 22
allow - udp - 22
# DNS
allow - tcp - 53
allow - udp - 53
```

```
while read line; do
        if [[ "$line" != *"#"* ]]; then
                activation=$(echo $line | cut -d '-' -f1 | tr -d '[:space:]')
                protocol=$(echo $line | cut -d '-' -f2 | tr -d '[:space:]')
                port=$(echo $line | cut -d '-' -f3 | tr -d '[:space:]')
                ${LOGGING} -i "Working on '$activation $port/$protocol'."
                ufw $activation $port/$protocol > /dev/null 2>&1
        fi
done < ${FW_CONF}
```

Berner Fachhochschule

DNS

NLNETLABS

unbound

NSD

Unbound is a validating, recursive, caching DNS resolver

NSD is an authoritative DNS name server

DNS

NLNETLABS

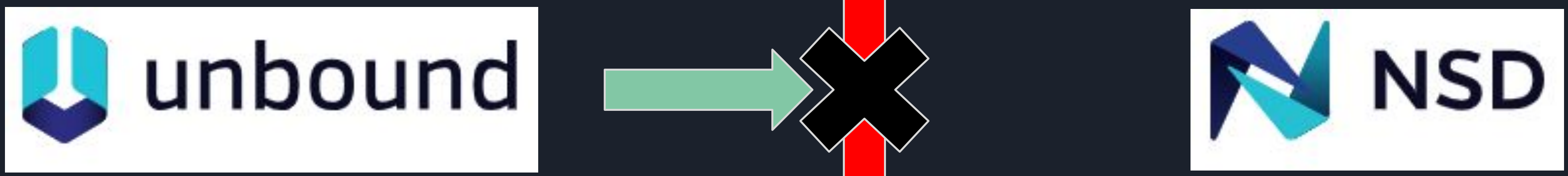
unbound

NSD

Unbound is a validating, recursive, caching DNS resolver

NSD is an authoritative DNS name server

DNS



Unbound is a validating, recursive, caching DNS resolver

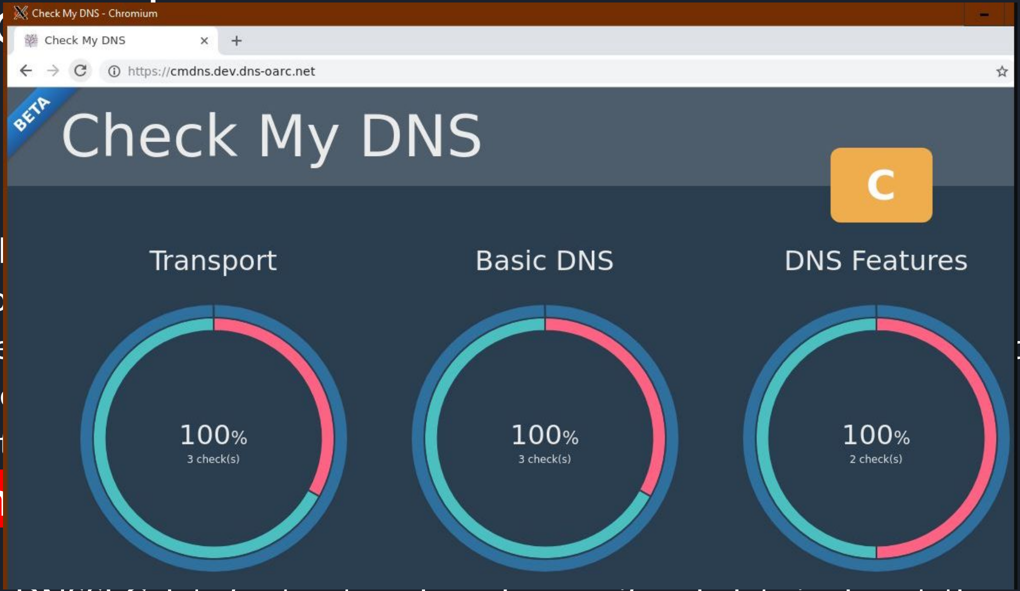
NSD is an authoritative DNS name server

Unbound

```
# Read the root hints from this file. Default is nothing, using built in
# hints for the IN class. The file has the format of zone files, with root
# nameserver names and addresses only. The default may become outdated,
# when servers change, therefore it is good practice to use a root-hints
# file. get one from https://www.internic.net/domain/named.root
root-hints: "/var/lib/unbound/root.hints"

# Require DNSSEC data for trust-anchored zones, if such data is absent, the
# zone becomes bogus. Harden against receiving dnssec-stripped data. If you
# turn it off, failing to validate dnskey data for a trustanchor will trigger
# insecure mode for that zone (like without a trustanchor). Default on,
# which insists on dnssec data for trust-anchored zones.
harden-dnssec-stripped: yes
```

Unbo



```
# Read tl
# hints fo
# namese                                                      d,
# when s
# file. get
```

root-h

```
# Require DNSSEC data for trust-anchored zones, if such data is absent, the
# zone becomes  bogus.  Harden against receiving dnssec-stripped data. If you
# turn it off, failing to validate dnskey data for a trustanchor will trigger
# insecure mode for that zone (like without a trustanchor).  Default on,
# which insists on dnssec data for trust-anchored zones.
```

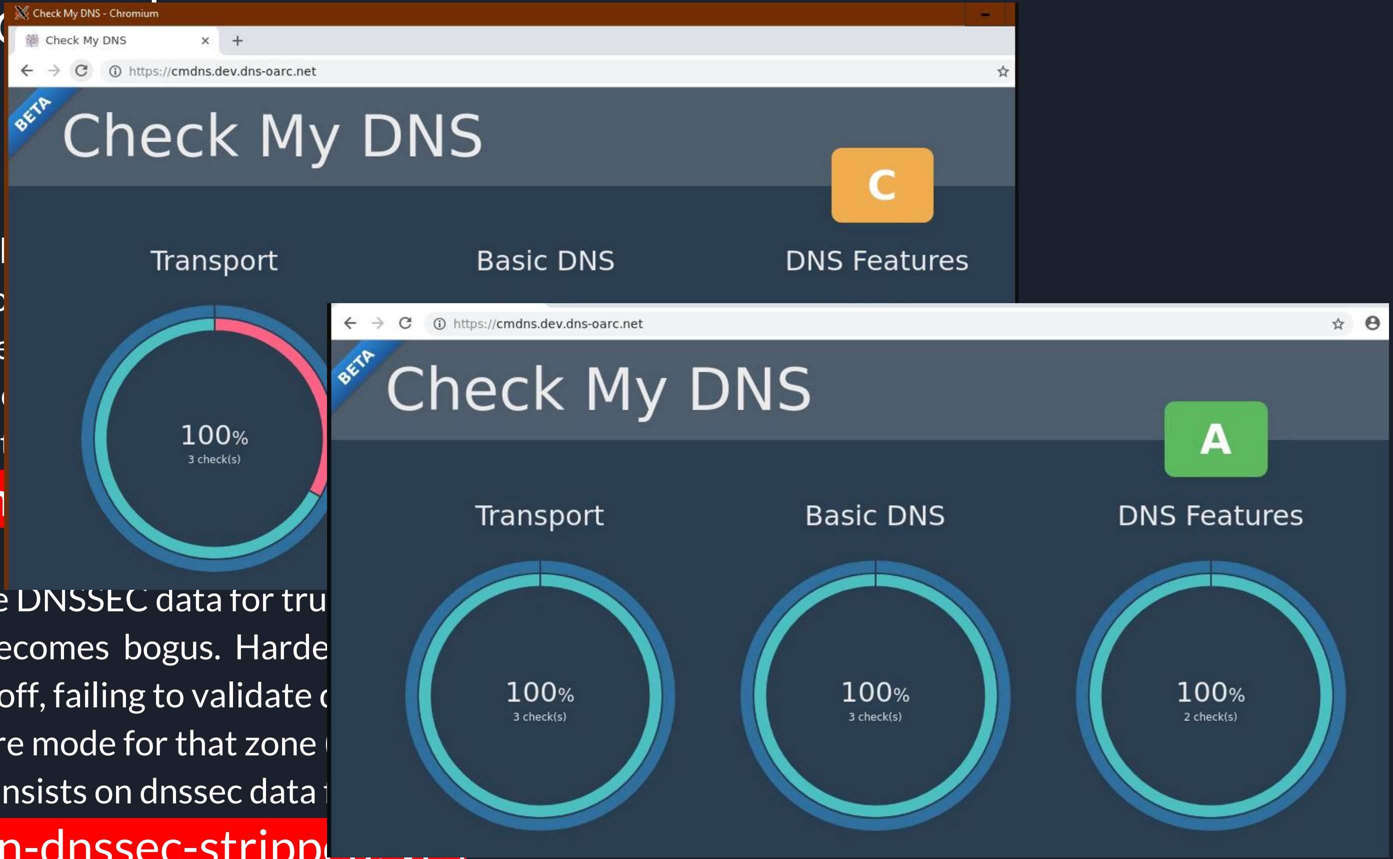
harden-dnssec-stripped: yes

Berner Fachhochschule

Unbo

```
# Read tl
# hints fo
# namese
# when s
# file. get
root-h

# Require DNSSEC data for tru
# zone becomes bogus. Harde
# turn it off, failing to validate
# insecure mode for that zone
# which insists on dnssec data
harden-dnssec-stripped: yes
```



Berner
Fachhochschule

NSD

```
$ORIGIN examplerun.cf.     ; default zone domain
$TTL    86400              ; default time to live


@ IN SOA ns1.examplerun.cf. ns2.examplerun.cf. (
        2019010917   ; serial number
        28800        ; Refresh
        7200         ; Retry
        1209600      ; Expire
        86400        ; Min TTL
        )
```

```
NS      ns1.examplerun.cf.
NS      ns2.examplerun.cf.
MX       10 mail.examplerun.cf.
```

```
examplerun.cf.    IN CAA 0 issue "letsencrypt.org"
examplerun.cf.    IN CAA 0 iodef
"mailto:postmaster@examplerun.cf"
```

```
  IN A  104.248.137.212
  IN TXT  "v=spf1 mx a ~all"
ns1   IN A  104.248.137.212
ns2   IN A  104.248.137.212
www    IN A  104.248.137.212
*     IN A  104.248.137.212
```

```
mail IN A 104.248.137.212
    IN TXT  "v=spf1 mx a ~all"
2019010917._domainkey      IN      TXT     (
"v=DKIM1\059 h=sha256\059 k=rsa\059 s=email\059
p="<DKIM KEY>" )
_adsp._domainkey       IN TXT "dkim=all"
_dmarc IN TXT "v=DMARC1\059 p=quarantine\059
sp=quarantine\059 adkim=r\059 aspf=r\059 fo=1\059
rf=afrf\059 rua=mailto:postmaster@examplerun.cf"
```

SSH

- SSH config hardening

```
X11Forwarding no
UseDNS yes
PermitRootLogin no
HostKeyAlgorithms
ssh-ed25519-cert-v01@openssh.com,ssh-rsa-cert-v01@openssh.com,ssh-ed25519,ssh-rsa,ecdsa-sha2-nistp521-cert-v01@
openssh.com,ecdsa-sha2-nistp384-cert-v01@openssh.com,ecdsa-sha2-nistp256-cert-v01@openssh.com,ecdsa-sha2-nistp5
21,  ecdsa-sha2-nistp384,ecdsa-sha2-nistp256

KexAlgorithms
curve25519-sha256@libssh.org,ecdh-sha2-nistp521,ecdh-sha2-nistp384,ecdh-sha2-nistp256,diffie-hellman-group-exch
ange-sha256

Ciphers
chacha20-poly1305@openssh.com,aes256-gcm@openssh.com,aes128-gcm@openssh.com,aes256-ctr,aes192-ctr,aes128-ctr
```

SSH

- Secure SSH Keys

```
ssh-keygen -b 4096 -C "$user@$DOMAIN" -E sha256 -N mys3cr3t -t rsa -f
/home/alice/.ssh/id_rsa
```

Mail - Postfix

Berner Fachhochschule

Mail - TLS

- TLS - Avoid clear text whenever possible

- DOVECOT
  - IMAP Server
  - Mutual Authentication

Berner
Fachhochschule

Mail - TLS

- Letsencrypt to obtain certificate

```
certbot certonly --agree-tos --standalone -n -m
postmaster@examplerun.cf -d mail.examplerun.cf
```

- Only use secure protocols (>TLSv1.2) and strong ciphers

```
smtpd_tls_mandatory_protocols = !SSLv2, !SSLv3, !TLSv1, !TLSv1.1
smtpd_tls_protocols = !SSLv2, !SSLv3, !TLSv1, !TLSv1.1
smtp_tls_mandatory_protocols = !SSLv2, !SSLv3, !TLSv1, !TLSv1.1
smtp_tls_protocols = !SSLv2, !SSLv3, !TLSv1, !TLSv1.1
smtp_tls_exclude_ciphers = EXP, MEDIUM, LOW, DES, 3DES, SSLv2
smtpd_tls_exclude_ciphers = EXP, MEDIUM, LOW, DES, 3DES, SSLv2
tls_high_cipherlist =
kEECDH:+kEECDH+SHA:kEDH:+kEDH+SHA:+kEDH+CAMELLIA:kECDH:+kECDH+SHA:kRSA:+kRSA+SHA:+kRSA+CAMELLIA:!aNULL:!eNULL:!SSLv2:!RC4:!MD5:!DES:!EXP:!S
EED:!IDEA:!3DES:!SHA'
tls_preempt_cipherlist   = yes
smtp_tls_ciphers = high
smtpd_tls_ciphers = high
```

Berner
Fachhochschule

Mail - Dovecot

- Require mutual authentication

```
ssl_verify_client_cert = yes
ssl_cert_username_field = CN
auth_ssl_username_from_cert = yes
```

- Map users

```
echo "alice: alice" >> /etc/aliases
echo "alice@example.cf alice@example.cf" >> /etc/postfix/canonical
echo "alice:::::::" >> /etc/dovecot/users-external
```

Mail - Anti-SPAM Measures

- SPF (Sender Policy Framework)
- DKIM (DomainKeys Identified Mail)
- ADSP (Author Domain Signing Practices)
- DMARC (Domain-based Message Authentication, Reporting and Conformance)

Mail - SPF

/etc/postfix/master.cf

```
 policyd-spf  unix  -       n       n       -       0       spawn
    user=policyd-spf argv=/usr/bin/policyd-spf
```

/etc/postfix/main.cf

```
 policyd-spf_time_limit = 3600
 smtpd_recipient_restrictions =
    ...
    reject_unauth_destination,
    check_policy_service unix:private/policyd-spf,
    ...
```

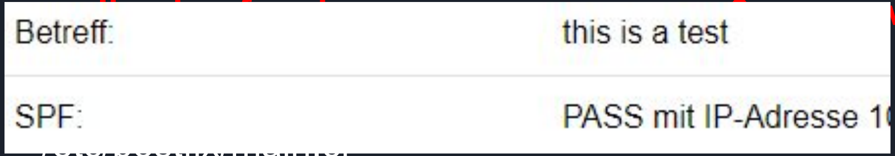
DNS zone file:

```
  IN TXT  "v=spf1 mx a ~all"
```

Mail - SPF

/etc/postfix/master.cf

wn



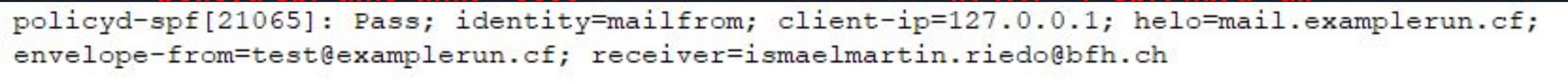
Betreff:	this is a test
SPF:	PASS mit IP-Adresse 1

/etc/postfix/main.cf

policyd-spf_time_limit = 3600

DNS zone file:

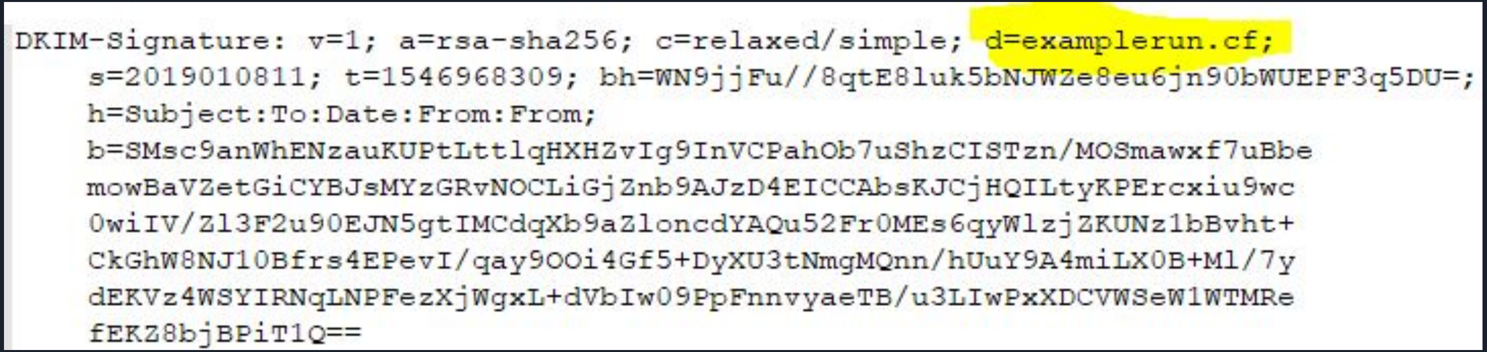
IN TXT "v=spf1 mx a ~all"



```
policyd-spf[21065]: Pass; identity=mailfrom; client-ip=127.0.0.1; helo=mail.examplerun.cf;
envelope-from=test@examplerun.cf; receiver=ismaelmartin.riedo@bfh.ch
```

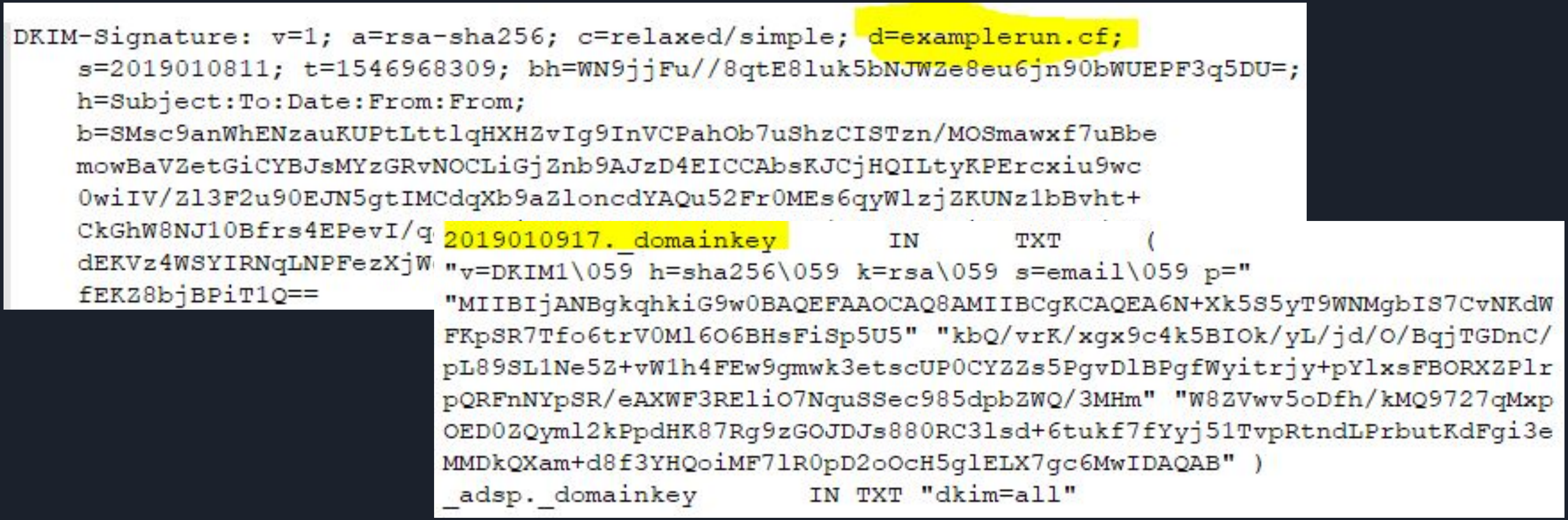
```
reject_unauth_destination,
check_policy_service unix:private/policyd-spf,
...
```

Mail - DKIM & ADSP



```
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=examplerun.cf;
    s=2019010811; t=1546968309; bh=WN9jjFu//8qtE8luk5bNJWZe8eu6jn90bWUEPF3q5DU=;
    h=Subject:To:Date:From:From;
    b=SMsc9anWhENzauKUPtLttlqHXHZvIg9InVCPahOb7uShzCISTzn/MOSmawxf7uBbe
    mowBaVZetGiCYBJsMYzGRvNOCLiGjZnb9AJzD4EICCAbsKJCjHQILtyKPErcxiu9wc
    0wiIV/Zl3F2u90EJN5gtIMCdqXb9aZloncdYAQu52Fr0MEs6qyWlzjZKUNz1bBvht+
    CkGhW8NJ10Bfrs4EPevI/qay9OOi4Gf5+DyXU3tNmgMQnn/hUuY9A4miLX0B+Ml/7y
    dEKVz4WSYIRNqLNPFezXjWgxL+dVbIw09PpFnnvyaeTB/u3LIwPxXDCVWSeW1WTMRe
    fEKZ8bjBPiT1Q==
```

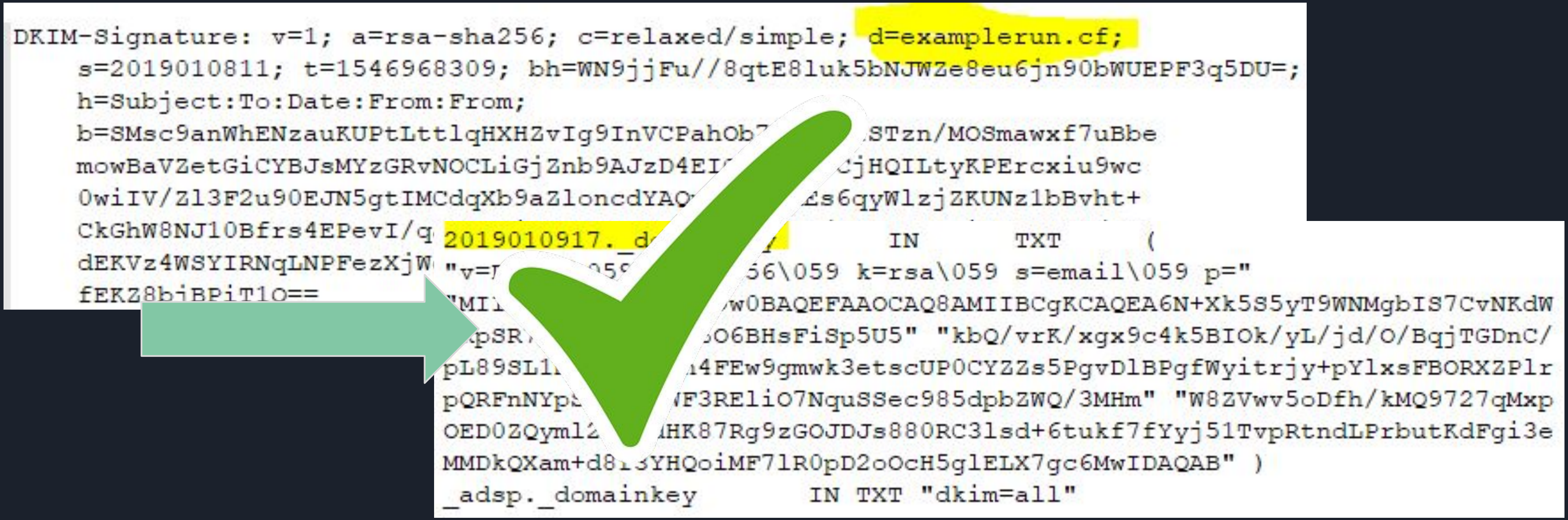
Mail - DKIM & ADSP



```
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=examplerun.cf;
    s=2019010811; t=1546968309; bh=WN9jjFu//8qtE8luk5bNJWZe8eu6jn90bWUEPF3q5DU=;
    h=Subject:To:Date:From:From;
    b=SMsc9anWhENzauKUPtLttlqHXHZvIg9InVCPahOb7uShzCISTzn/MOSmawxf7uBbe
    mowBaVZetGiCYBJsMYzGRvNOCLiGjZnb9AJzD4EICCAbsKJCjHQILtyKPErcxiu9wc
    0wiIV/Zl3F2u90EJN5gtIMCdqXb9aZloncdYAQu52Fr0MEs6qyWlzjZKUNz1bBvht+
    CkGhW8NJ10Bfrs4EPevI/q
    dEKVz4WSYIRNqLNPFezXjW
    fEKZ8bjBPiT1Q==
```

```
2019010917._domainkey         IN       TXT      (
"v=DKIM1\059 h=sha256\059 k=rsa\059 s=email\059 p="
"MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA6N+Xk5S5yT9WNMgbIS7CvNKdW
FKpSR7Tfo6trV0Ml6O6BHsFiSp5U5" "kbQ/vrK/xgx9c4k5BIOk/yL/jd/O/BqjTGDnC/
pL89SL1Ne5Z+vW1h4FEw9gmwk3etscUP0CYZZs5PgvDlBPgfWyitrjy+pYlxsFBORXZPlr
pQRFnNYpSR/eAXWF3REliO7NquSSec985dpbZWQ/3MHm" "W8ZVwv5oDfh/kMQ9727qMxp
OED0ZQyml2kPpdHK87Rg9zGOJDJs880RC3lsd+6tukf7fYyj51TvpRtndLPrbutKdFgi3e
MMDkQXam+d8f3YHQoiMF7lR0pD2oOcH5glELX7gc6MwIDAQAB" )
_adsp._domainkey       IN TXT "dkim=all"
```

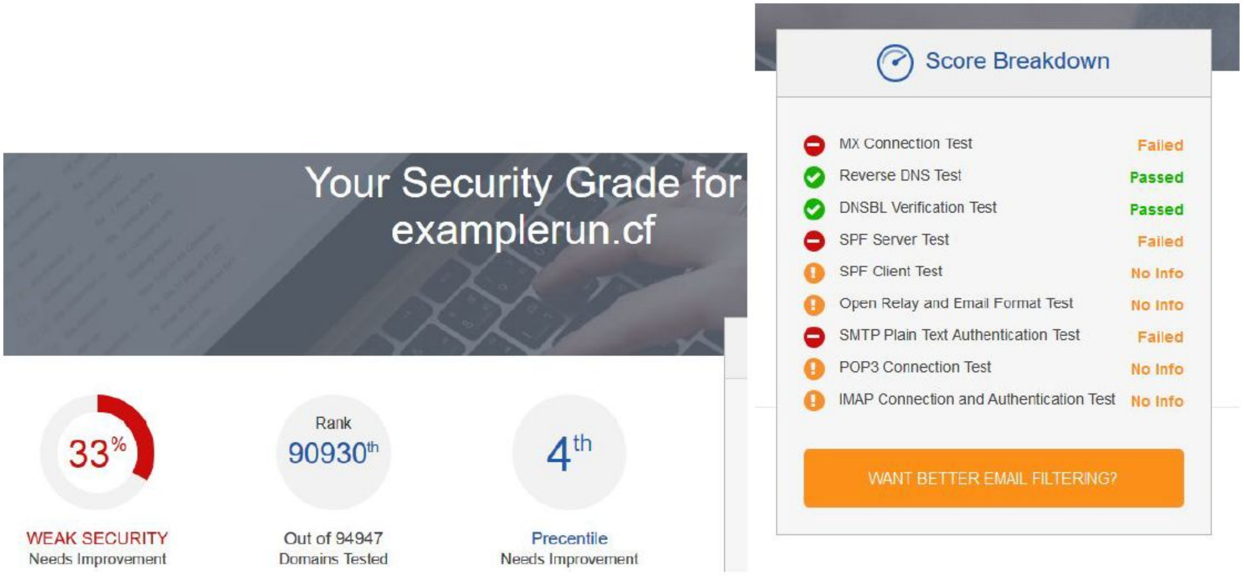
Mail - DKIM & ADSP



```
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=examplerun.cf;
    s=2019010811; t=1546968309; bh=WN9jjFu//8qtE8luk5bNJWZe8eu6jn90bWUEPF3q5DU=;
    h=Subject:To:Date:From:From;
    b=SMsc9anWhENzauKUPtLttlqHXHZvIg9InVCPahOb7        STzn/MOSmawxf7uBbe
    mowBaVZetGiCYBJsMYzGRvNOCLiGjZnb9AJzD4EI          CjHQILtyKPErcxiu9wc
    0wiIV/Zl3F2u90EJN5gtIMCdqXb9aZloncdYAQ          Zs6qyWlzjZKUNz1bBvht+
    CkGhW8NJ10Bfrs4EPevI/q
    dEKVz4WSYIRNqLNPFezXjW
    fEKZ8bjBPiT1O==
```

```
2019010917._d                    IN        TXT       (
"v=D       059              56\059 k=rsa\059 s=email\059 p="
"MI               w0BAQEFAAOCAQ8AMIIBCgKCAQEA6N+Xk5S5yT9WNMgbIS7CvNKdW
pSR               O6BHsFiSp5U5" "kbQ/vrK/xgx9c4k5BIOk/yL/jd/O/BqjTGDnC/
pL89SL1           4FEw9gmwk3etscUP0CYZZs5PgvDlBPgfWyitrjy+pYlxsFBORXZPlr
pQRFnNYp          F3RElio7NquSSec985dpbZWQ/3MHm" "W8ZVwv5oDfh/kMQ9727qMxp
OED0ZQyml2        HK87Rg9zGOJDJs880RC3lsd+6tukf7fYyj51TvpRtndLPrbutKdFgi3e
MMDkQXam+d8       3YHQoiMF7lR0pD2oOcH5glELX7gc6MwIDAQAB" )
_adsp._domainkey        IN TXT "dkim=all"
```

Mail - Postfix



Your Security Grade for examplerun.cf

33%
WEAK SECURITY
Needs Improvement

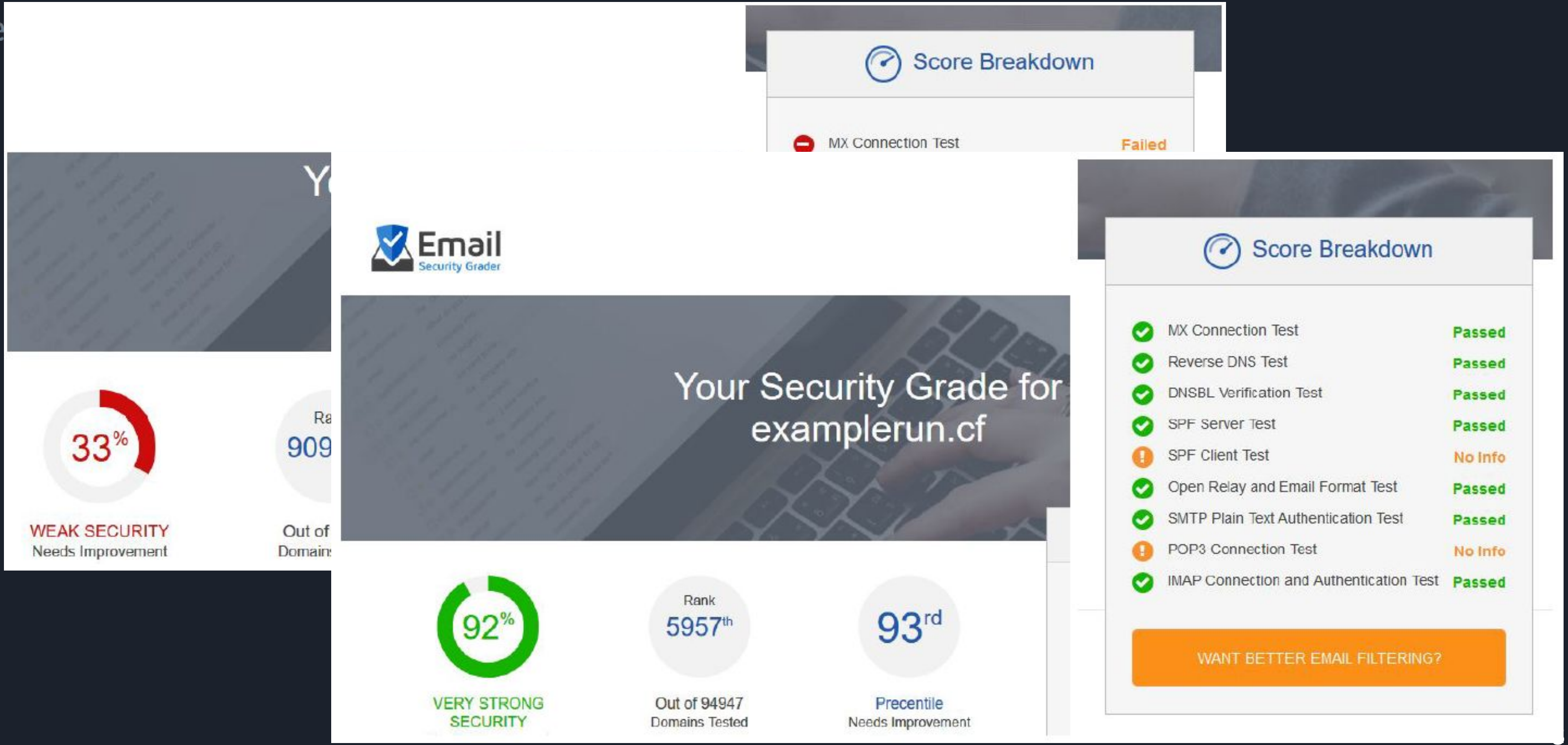
Rank
90930th
Out of 94947
Domains Tested

4th
Precentile
Needs Improvement

Score Breakdown

MX Connection Test	Failed
Reverse DNS Test	Passed
DNSBL Verification Test	Passed
SPF Server Test	Failed
SPF Client Test	No Info
Open Relay and Email Format Test	No Info
SMTP Plain Text Authentication Test	Failed
POP3 Connection Test	No Info
IMAP Connection and Authentication Test	No Info

Mail - Postfix



Score Breakdown

MX Connection Test Failed

33% WEAK SECURITY Needs Improvement

Ra 909

Out of Domain

Email Security Grader

Your Security Grade for examplerun.cf

92% VERY STRONG SECURITY

Rank 5957th Out of 94947 Domains Tested

93rd Precentile Needs Improvement

Score Breakdown

MX Connection Test	Passed
Reverse DNS Test	Passed
DNSBL Verification Test	Passed
SPF Server Test	Passed
SPF Client Test	No Info
Open Relay and Email Format Test	Passed
SMTP Plain Text Authentication Test	Passed
POP3 Connection Test	No Info
IMAP Connection and Authentication Test	Passed

WANT BETTER EMAIL FILTERING?

Berner
Fachhochschule

Challenges

Major Event

- From Federated Project to Consolidated Project

Setup

- Shared responsibility common entry point

KISS (Keep It Simple and Stupid)

- Use your own knowledge to not to use it

Berner
Fachhochschule

DEMO

All documents and code can be found at :

https://github.com/ifrido/BTI7301

END

Web - Optional Part

src/web/apache/
- enableApache.sh
- configureApache.sh

src/web/nginx/
- enableNginx.sh
- nginxCertConfig.sh
- configureNginx.sh

All documents and code can be found at :

https://github.com/ifrido/BTI7301

END