Development and Security Analysis of pVPKIweb, a Python API and a Web Interface for NSS VPKI*

Adnan Shafi
EP2520 BNSS, Group 9
ashafi@kth.se

Acknowledgement

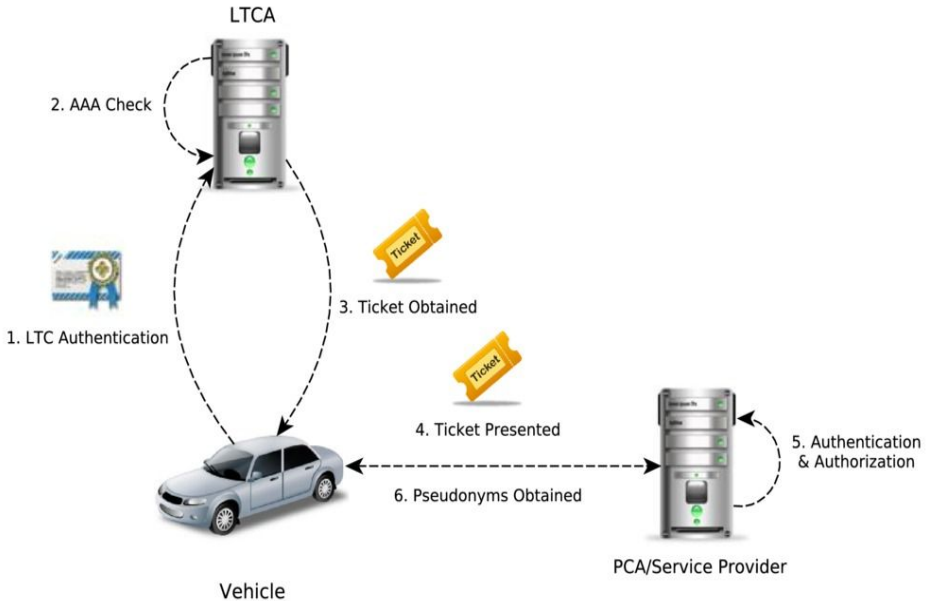
- M. Khodaei, H. Jin, and P. Papadimitratos, “Towards Deploying a Scalable & Robust Vehicular Identity and Credential Management Infrastructure,” in IEEE VNC, Paderborn, Germany, Dec. 2014.
- M. Khodaei and P. Papadimitratos, “Evaluating On-demand Pseudonym Acquisition Policies in Vehicular Communication Systems,” in Proceedings of the First International Workshop on Internet of Vehicles and Vehicles of Internet, Paderborn, Germany, pp. 7–12, July 2016.
- OWASP attacks seperately described using OWASP Top 10 slides from owasp.org for year 2013. Not included in this file.

Agenda

- Background NSS VPKI
- What is this project about?
  - Part 1 - develop python API for VPKI
  - Part 2 - developed Web GUI based on above python API for VPKI
- Functional Testing
- Security Testing based on OWASP
- Conclusion

Background - What is NSS VPKI?

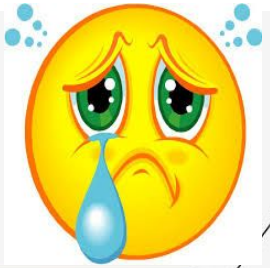
- Introduction
- Key Features
  - Privacy Preservation
- Components
- Terminology & Use Cases
  - Voucher
  - LTCA
  - Ticket
  - PCA
- Protocols
  - Security: IEEE 1609.2 standard
  - Elliptic Curve Cryptography



[M.Khodaei, H. Jin & P. Papadimitratos]

VPKI API - Abstraction Challenge

from here



msgPsnymCertReq_V2PCA
- iReqType
- iTicketSize
- strTicket
- iLTCAIdRange
- iPCAIdRange
- iLocation
- uiPsnymCertNo
- msgWAVECertificateRequest
- iNonce
- tTimeStamp

msgWAVECertificateRequest
- msgSignerInfo
- msgToBeSignedCSR
- msgSignature

msgSignerInfo
- msgSignerIdentifierType.SignerIdentifierType
- strCertificate
- strDigest
- strCertificatesChain

msgToBeSignedCSR
- csrVersion
- msgSubType.SubjectType subjectType
- msgRequestScopeType.RequestScopeType
- msgCertSpecificData
- msgECPublicKey

msgSignature
- uiCurveOrderOctets
- uiSignLen
- strSignature

msgECPublicKey
- uiPsnymPublicKeyLen
- strPsnymPublicKey

protobuf
Protocol Buffers

Base64

XML RPC

https://

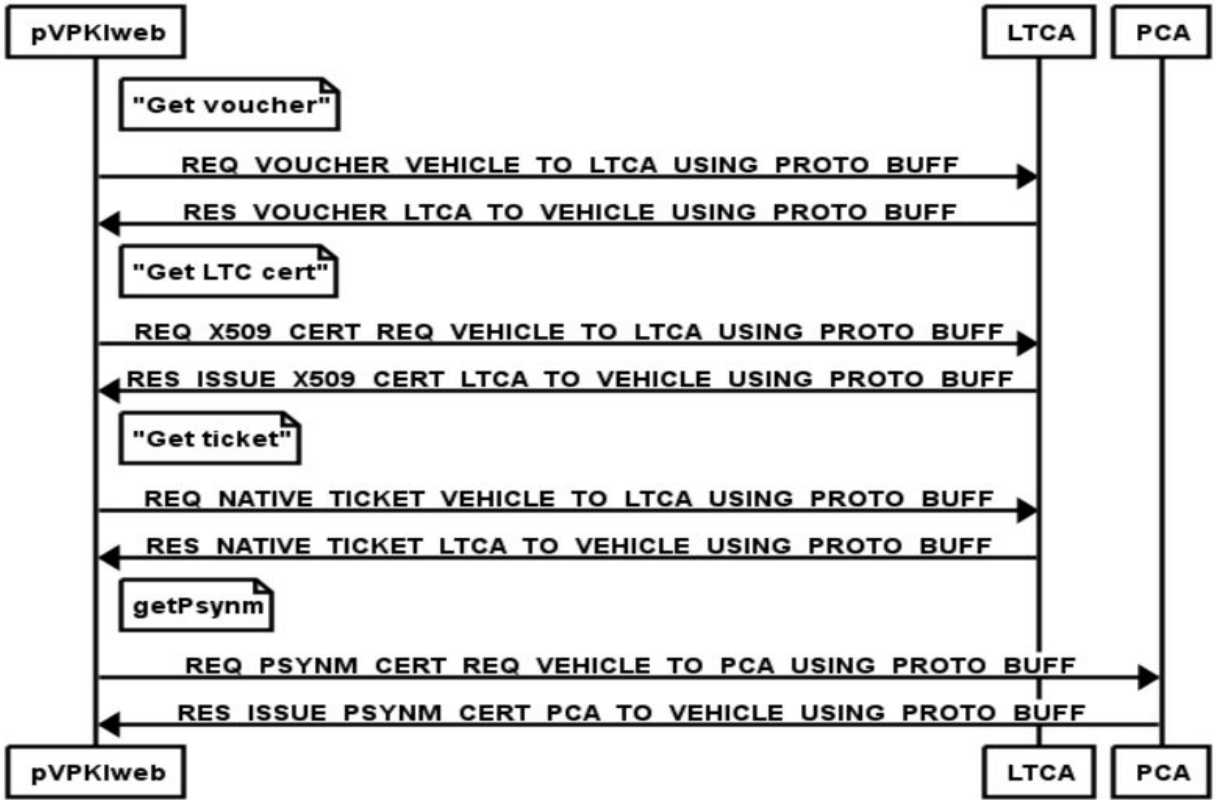
Crypto APIs

SHA-256

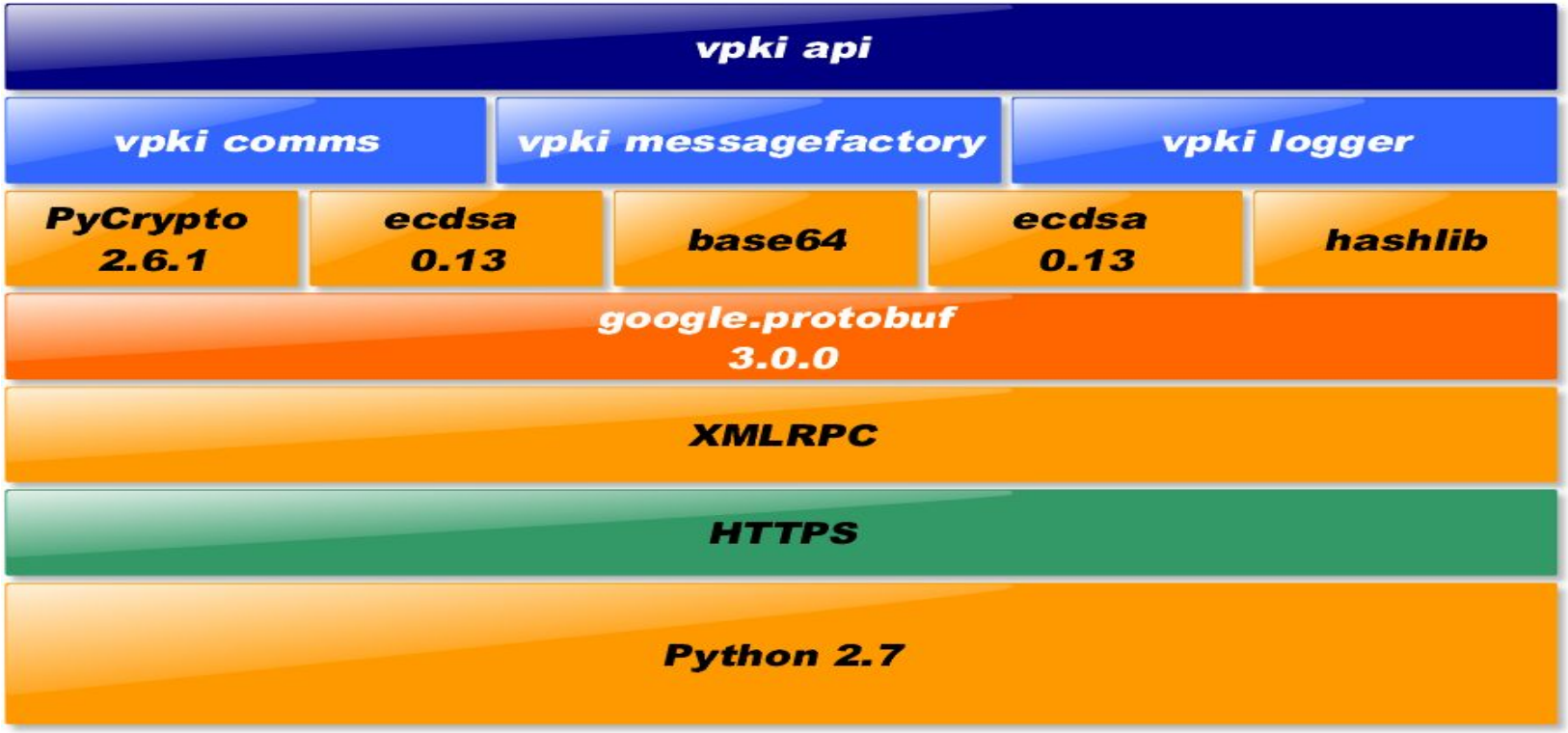
to there

get_psynm(..)

VPKI use cases



Python API for VPKI - Architecture description



VPKI Python API stack

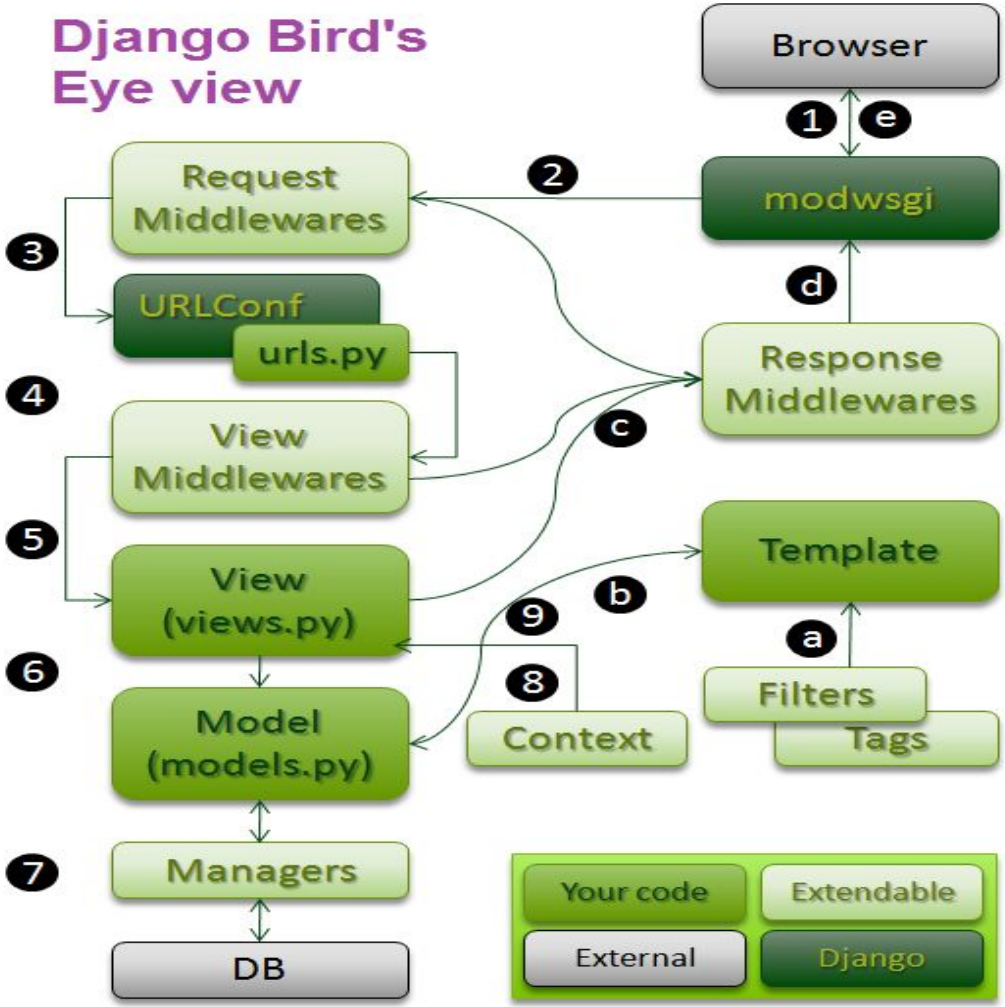
Python API - Issues and pitfalls!

- Lack of specification of the data structures in interface.proto file. Reliance on verbal input.
  - Lots of trial and error at times leading to unexpected time losses.
- Stability of crypto API is more important at times than ease of use.
- Force elliptic curve API to generate hashes encoded in Distinguished Encoding Rules (DER), while keys should be Privacy Enhanced Mail (PEM) encoded. Defaults won't work.
- Mind the :
  - elliptic curve NIST profiles. E.g. VPKI uses NIST256p
  - HTTP encoding e.g. UTF8.
- Do not forget to ASCII armour i.e. base64 encode before sending out.

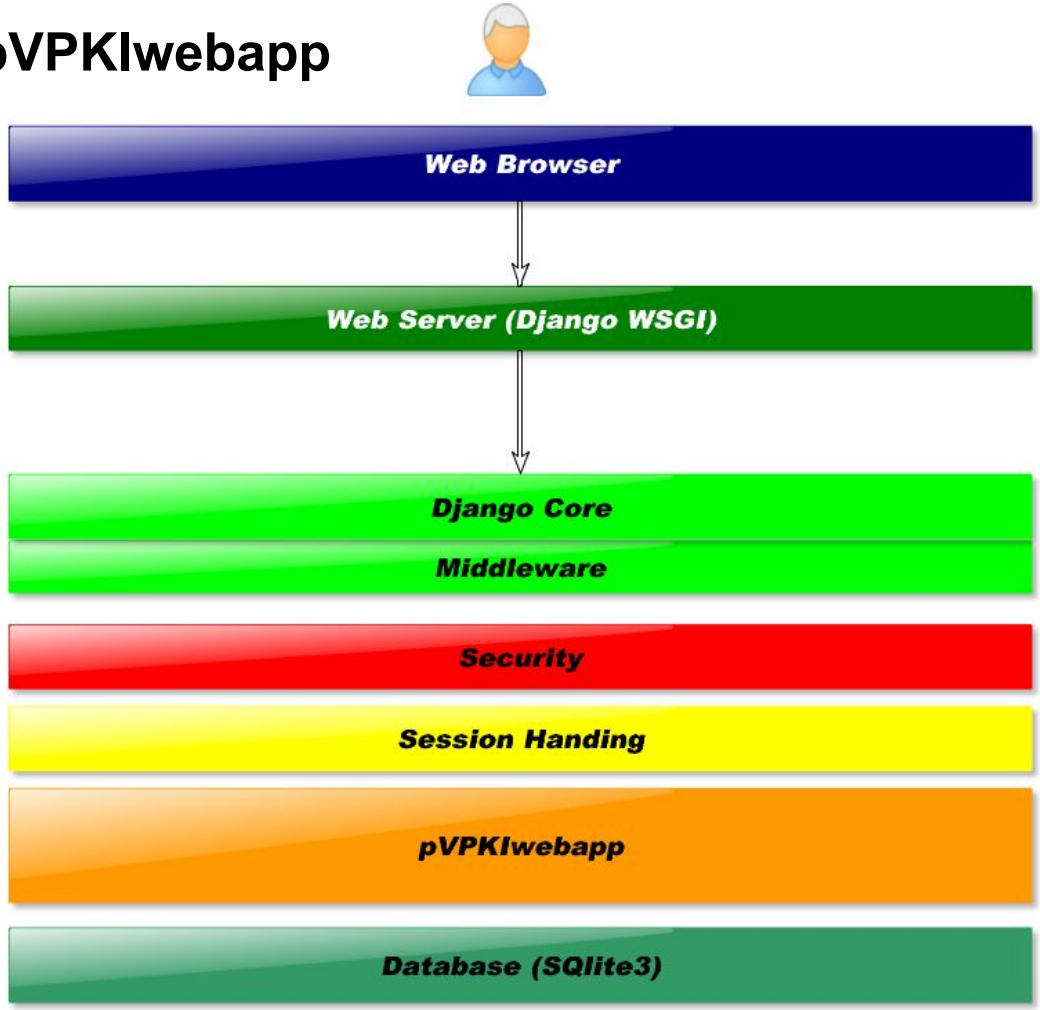
Web GUI for VPKI - pVPKIweb

- Uses Python API developed for VPKI to provide a web interface
- Uses Django web framework

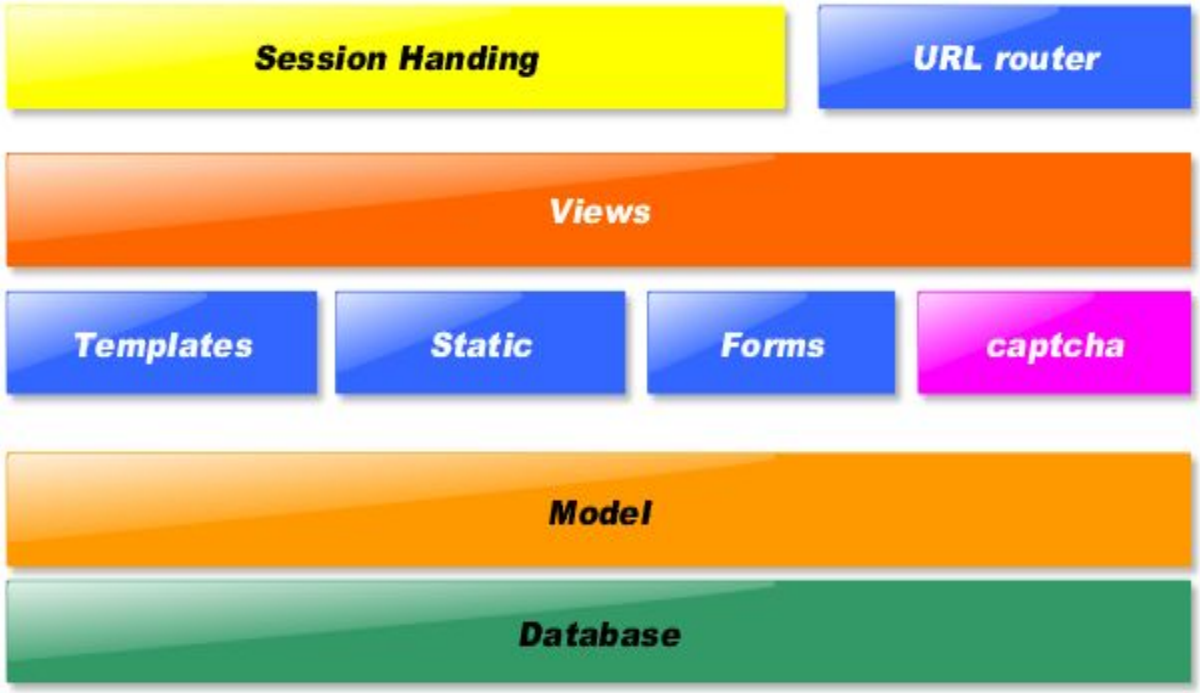
[Tesla Model S, www.tesla.com]


Django Bird's Eye view
Browser
1
e
Request Middlewares
2
modwsgi
3
URLConf
urls.py
d
Response Middlewares
4
View Middlewares
c
5
Template
View (views.py)
b
9
a
6
8
Filters
Tags
Model (models.py)
Context
7
Managers
DB
Your code
Extendable
External
Django

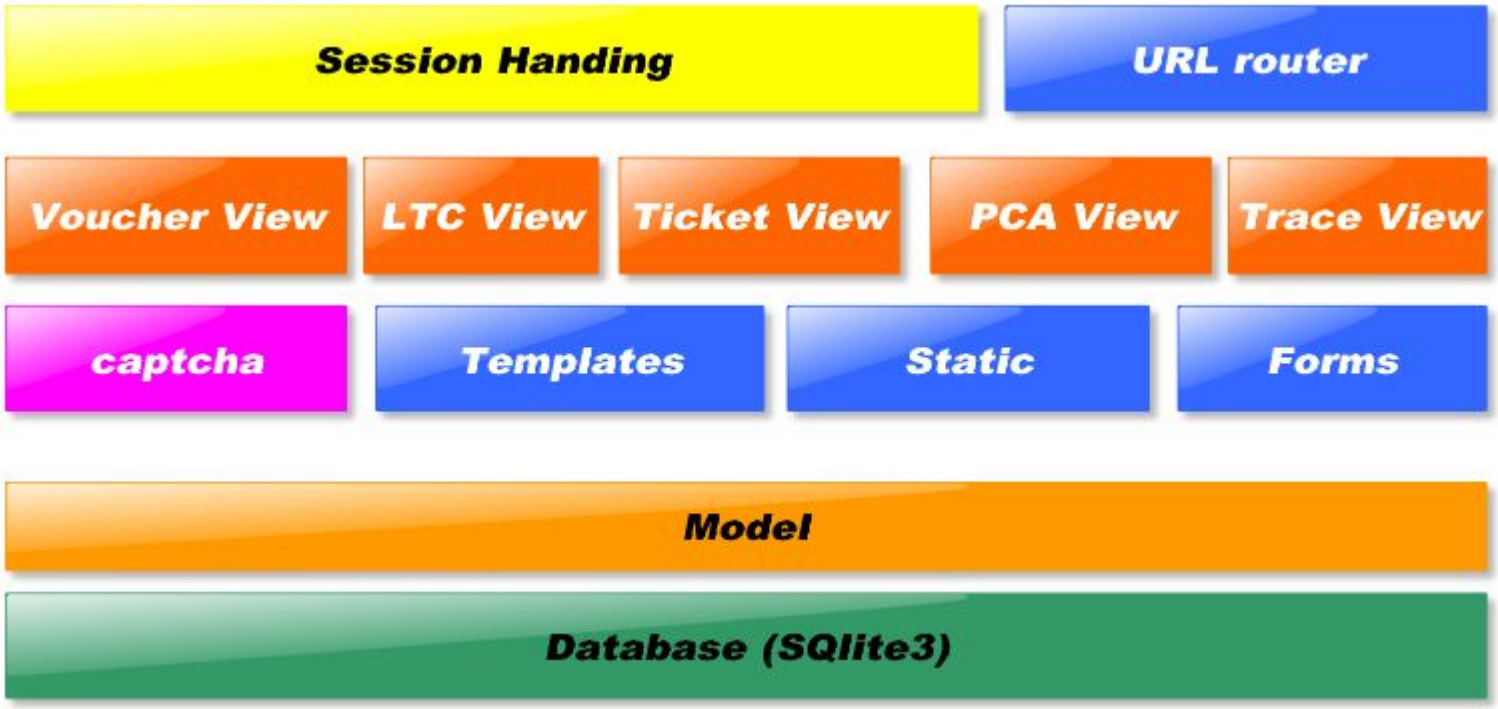
Architecture pVPKIwebapp



pVPKIwebapp Internals



pVPKIwebapp Internals : Low Level View



Function Testing

- Selenium
  - Driver for firefox and chrome on linux
  - Python unittest
- Avoiding email storm via captcha
  - Django-simple-captcha
- Testing on mobile phones and tablets
  - For iOS:
    - Emulator: appteize.io
    - Physical: iPAD mini
  - For android:
    - Physical: Samsung S4
- An integrated approach, more later.

“You cannot build secure web applications unless you know how they will be attacked”

“This was fine for your nephew’s fifth, Sire, but I fear it is set for a sterner test.”

Thanks to Royston Robertson www.roystonrobertson.co.uk for permission to use his cartoon

Is my Application vulnerable?

- Perform “Application Assessment” to find vulnerabilities
  - Vulnerability scanning
  - Code review
  - Penetration Testing
  - Static Analysis

Open Web App. Security Project (OWASP)

- Introduction
- Application Security
- OWASP Top 10
- Proactive Controls

OWASP Top10 from 2013

[OWASP.ORG]

Lab work server side

- WebGoat
  - To study real vulnerabilities, hands-on.
  - Version 7.1 latest
  - Java -jar webogat7.1-exec.jar; firefox localhost:8080
- Vulnhub's BWAPP-BEE-BOX
  https://www.vulnhub.com/entry/bwapp-bee-box-v16,53/
- OWASP Broken Web Applications Project
  - **Download OWASP_Broken_Web_Apps_VM_1.2.7z (1.8 GB)**
- Tools
  - OWASP ZAP

Lab work client side

Firefox addons

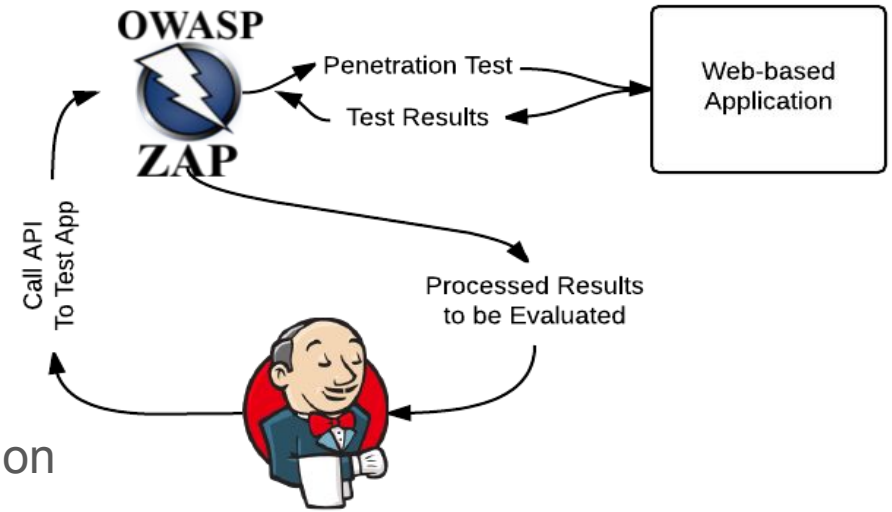
- OWASPmantra
  - For firefox, apt-get install owasp-mantra-ff
  - For chrome, http://www.getmantra.com/mantra-on-chromium.html
- Cookies Manager+
- Firebug
- Hackbar
- Http Requester
- Passive Recon

Setup

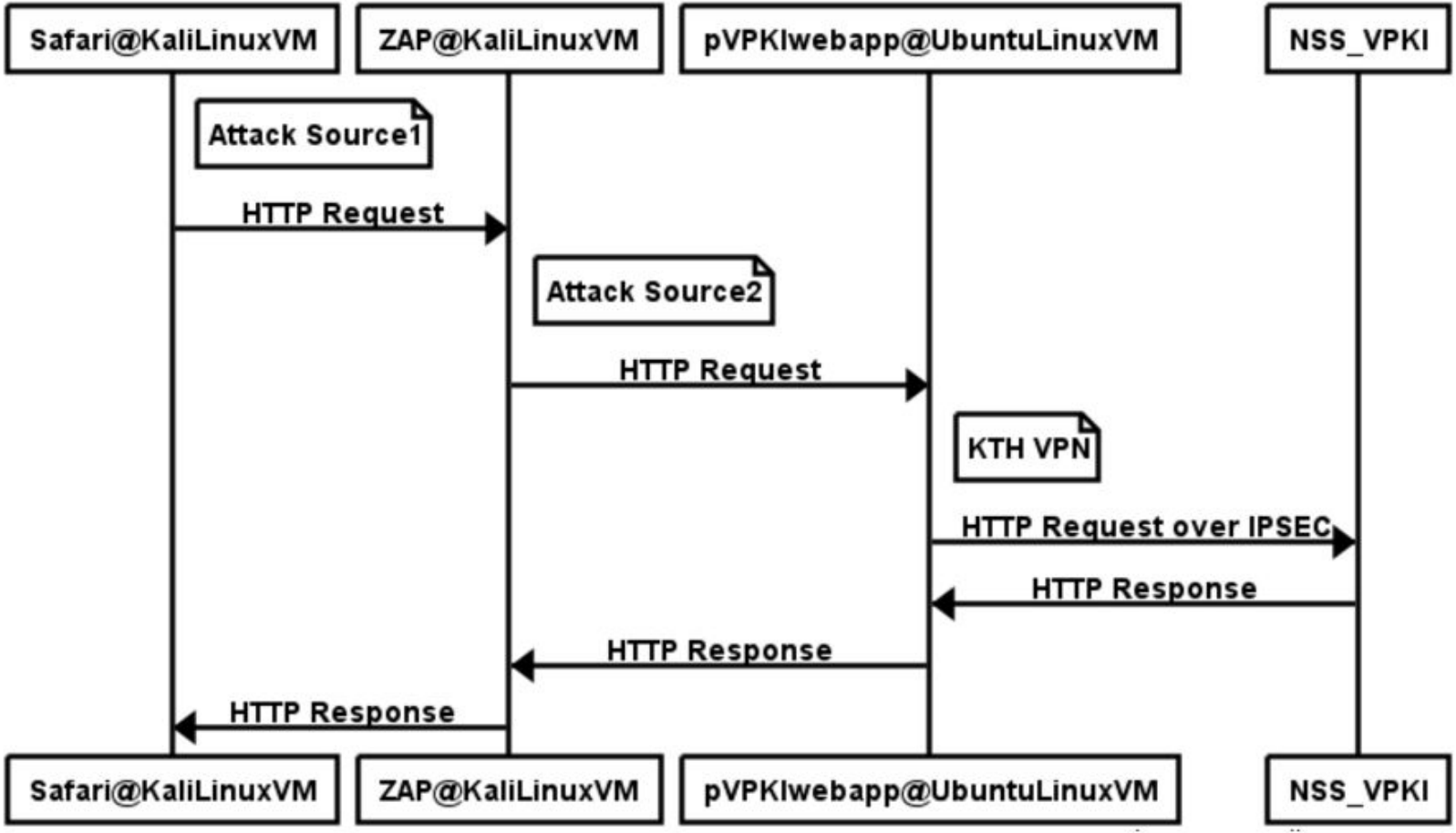
- Virtualbox running Ubuntu Linux VM
  - Running the pVPKIwebapp
  - Via KTH VPN connected
- Virtualbox running latest Kali Linux
  - Running firefox with extra client side plugins installed
  - Running interception proxy and other tools
- Tools
  - E.g.an interception proxy : Zed Attack Proxy (ZAP)

ZED Attack Proxy (ZAP) - an interception proxy

- Modes
  - Passive
  - Active
- Attacks
  - CSRF
  - XSS
  - Input Validation
    - Fuzzing
- Optional ZAP Python API
- Optional Selenium integration

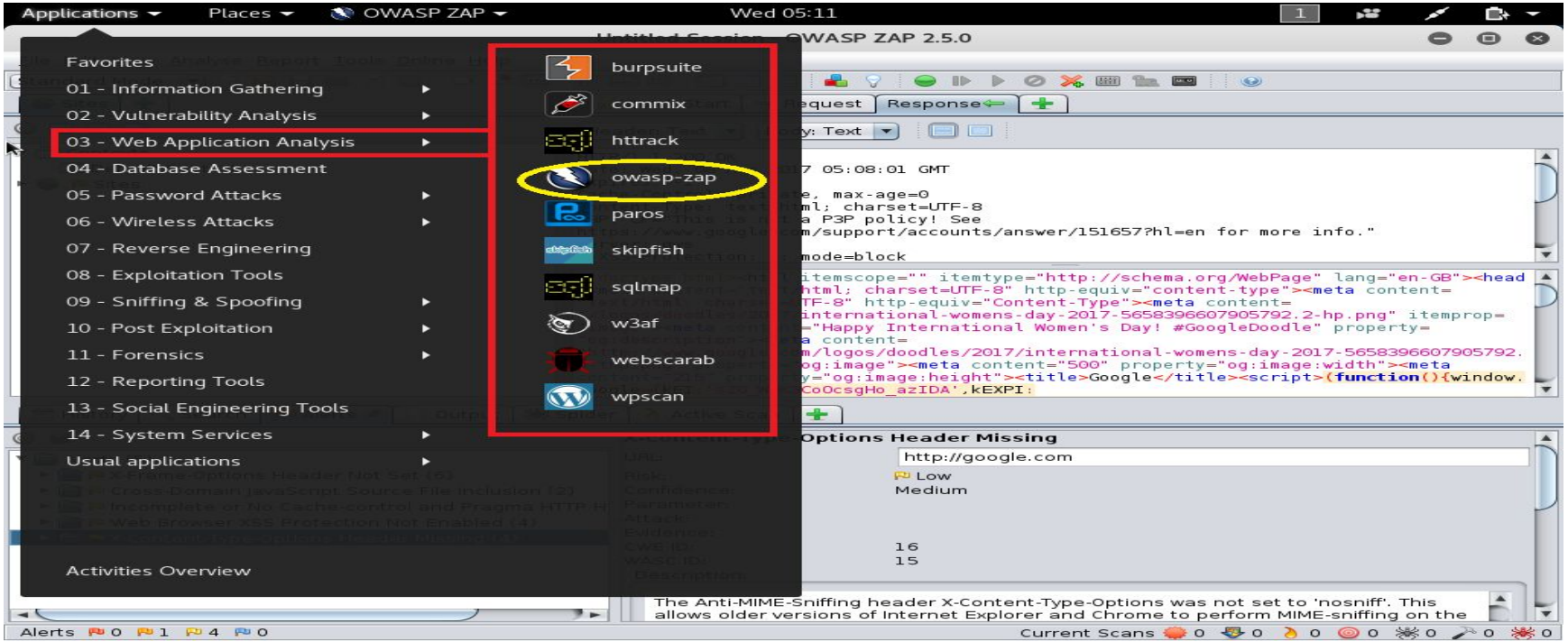


Security Testing Setup



ZAP : analysis of results

Explained during the demo along with the analysis of the ZAP results.



Future work

- Implement other NSS VPKI use cases e.g. roaming into a foreign domain.
- Depending on how the web gui should be used, implement additional security countermeasures e.g
  - Secure handling and storage for cryptographic keys
- Based on the python API, create a test bed for experimenting with VPKI. VPKI Emulation Lab.
  - V2V experimentation and proof of concepts

Demo

- Python API
- Web GUI
- Security Testing

Thankyou for listening!

Questions