<!-- Slide number: 1 -->
# OWASP Security Research and Development Framework
Anwar Mohamed
anwarelmakrahy@gmail.com
Twitter: @anwarelmakrahy
April 12th, 2014

### Notes:

<!-- Slide number: 2 -->
Agenda
Background
Introduction
Why SRDF?
Design Overview
Features
Python Wrapper
Demo
Projects based on SRDF

![](Picture2.jpg)
2

### Notes:

<!-- Slide number: 3 -->
Background
Anwar Mohamed
Computer & Communications Under-graduate.
Focusing on Android Security.
Author or Packetyzer, Whatsapp & Viber dump tools for Metasploit and others.
Co-Author of SRDF.
Amr Thabet
Malware Researcher at Q-CERT.
Wrote a Stuxnet Malware Analysis Paper .
Author of Pokas x86 Emulator.
Author of SRDF.

3

### Notes:

<!-- Slide number: 4 -->
Introduction
Development Framework (Library).
Contains many security classes/tools.
Created For:
Malware Analysis.
Packet Analysis.
Antivirus and Firewall Tools.
Free and Open Source.
4

### Notes:

<!-- Slide number: 5 -->

Why SRDF?

![](Picture2.jpg)
5

### Notes:

<!-- Slide number: 6 -->
Why SRDF?
Implement your Innovative Idea.
Don’t re-invent The Wheel.
Don’t waste your time.
Flexible Design.
Production Quality.
Community Based Development and Beta-testing.

6

### Notes:

<!-- Slide number: 7 -->
Design OverviewUser-Mode Design

![](Picture2.jpg)
7

### Notes:

<!-- Slide number: 8 -->
Design OverviewUser-Mode Design
Infrastructure:
This includes the essential elements of any development framework and it’s not related to security, like: string, hash, list, serializer, database, registry manipulation, sockets and so on.
Targets:
This is the beginning of SRDF. This part is simply the Target from your security tool. What do you want to secure or be secured from. It includes Files (PE Files and others), Processes and Packets.
Libraries:
Malware: includes the assemblers and disassemblers, emulator, debugger, API Hooker, Yara Scanner (wildcard scanner) file recursive scanner and other tools.
Network: includes User-Mode capturing and Firewall.
Core (The Application Interface):
The Core includes the Logging system and the back-end Database.
8

### Notes:

<!-- Slide number: 9 -->
Features
Based on OOP.
PE, ELF, PDF and Dex File Parsers.
x86 Disassembler, Debugger and Emulator.
API Hooking.
Packet, Protocol and Network Flow Analysis.
Production Quality.
FREE and Open Source.

9

### Notes:

<!-- Slide number: 10 -->
Python Wrapper
pySRDF
Python implementation for The Security Research and Development Framework Project.
That's the strongest reverse engineering and malware analysis tool for python and the easiest to install and use .
https://github.com/AmrThabet/pySRDF

10

### Notes:

<!-- Slide number: 11 -->
DemoPE Analyzer

![](Picture2.jpg)

![](Picture3.jpg)
11

### Notes:

<!-- Slide number: 12 -->
DemoProcess Analyzer

![](Picture2.jpg)

![](Picture3.jpg)
12

### Notes:

<!-- Slide number: 13 -->
DemoPacket Analyzer

![](Picture2.jpg)
13

### Notes:

<!-- Slide number: 14 -->
Projects based on SRDF
Inspector’s Gadget.
Exploitation Detection System.
httpcat Tool.

14

### Notes:

<!-- Slide number: 15 -->
How to Contribute?
winSRDF
https://github.com/AmrThabet/winSRDF
pySRDF
https://github.com/AmrThabet/pySRDF
SRDF on OWASP
https://www.owasp.org/index.php/OWASP_Security_Research_and_Development_Framework

15

### Notes:

<!-- Slide number: 16 -->
Thank You
Any Questions?
16

### Notes: