Hack.lu edition 2012
A forensic analysis of Android Malware

Wednesday, 24th October 2012

KEVIN ALLIX, kevin.allix@uni.lu
QUENTIN JEROME, quentinjerome939@gmail.com
RADU STATE, radu.state@uni.lu
JACQUES KLEIN, jacques.klein@uni.lu

SnT, Université du Luxembourg

Be scared of Malware!

securityandtrust.lu

Android Malware

- At least one story in the news every single week
- A target of choice because of the amount of data stored on smartphones

According to Antivirus vendors. . .

- You should be scared
- You should buy their product

The Starting Point

securityandtrust.lu

A dataset of Android Malware

- 1258 malware
- `http://www.malgenomeproject.org`
- *Dissecting Android Malware: Characterization and Evolution*
  Yajin Zhou, Xuxian Jiang
  Proceedings of the 33rd IEEE Symposium on Security and Privacy (Oakland 2012) San Francisco, CA, May 2012

A dataset of clean Android Applications

- Collected from Google Play...
- ...and from alternative Markets

Crawling for apps... I

securityandtrust.lu

Several markets

Google play (Android Market)	The official market
AppChina	The biggest alternative market
Slideme	U.S. Based market, free and paid apps
FreewareLovers	German company, Works with a browser!
ProAndroid	Russian market for free apps
Torrents	BitTorrent seems to be a popular way of getting apps...
Genome	The Malware dataset

Crawling for apps... II

Markets way not want to be scraped :(

- Google play uses an overly complex protocol, with authentication and scraping counter-measures (even for free apps)
- AppChina enforces drastic scraping protections: 1Mb/s bandwidth limitation, several-hour ban if using more than one connection

Crawling for apps... III

securityandtrust.lu

What we collected

Table: Number of Apps for each market

Market name	**App Number**
Google	51,885
AppChina	63,476
Slideme	15,409
FreewareLovers	4,119
ProAndroid	2,633
Torrents	5,255
Genome	1,258
total	144,035
total (dedup)	137,105

The Metadata Approach I

securityandtrust.lu

Android Applications

- Distributed as Single-file `.apk` Packages
- Just a `zip` file that contains:
  - Compiled Code (Dalvik Bytecode)
  - A Manifest file
  - Resources (images, sounds, . . .)
  - One Certificate (or more than one)
  - A signature file

$\rightarrow$ Apps travel unmodified from Authors' computer to users' smartphones. No tinkering done by Markets

The Metadata Approach II

securityandtrust.lu

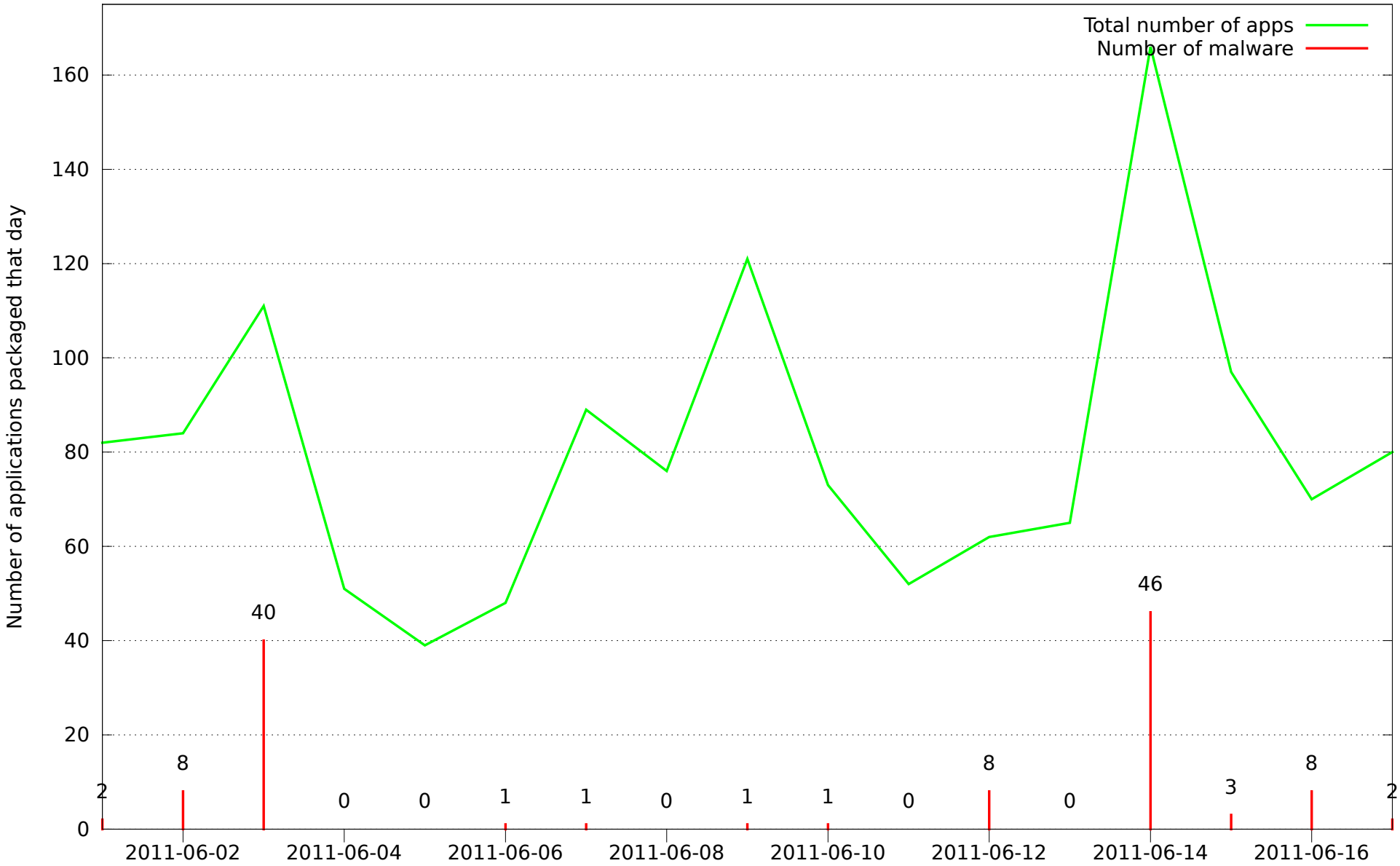
Questions

- Can we find some patterns in the metadata?
- Can we find *different* patterns for Malware?

We Focus on two metadata:

- Packaging Timestamp
- Certificate

Timestamps


Total number of apps
Number of malware
Number of applications packaged that day
0
20
40
60
80
100
120
140
160
2011-06-02
2011-06-04
2011-06-06
2011-06-08
2011-06-10
2011-06-12
2011-06-14
2011-06-16
2
8
40
0
0
1
1
0
1
1
0
8
0
46
3
8
2

Timestamps: Fun Facts I

securityandtrust.lu

Malware come in batches

- The 1258 known Malware were packaged on only 244 different days, while they cover $\sim$ 3 years
- Record day: 2011-09-21 (51 malware, nearly 25% of all apps that day)
- Only 72 malware (5.7%) were packaged a day when no other malware was packaged.

Timestamps: Fun Facts II

securityandtrust.lu

Malware Authors can do much better

- We counted 78 cases where at least two malware were packaged in the same second
- At 15 instances, four or more malware were packaged in the same second
- Two of those instances saw ten or more new malware being packaged.

This strong time locality suggests that malware writers have set up tools to automate the malware packaging process

Distribution over days of week

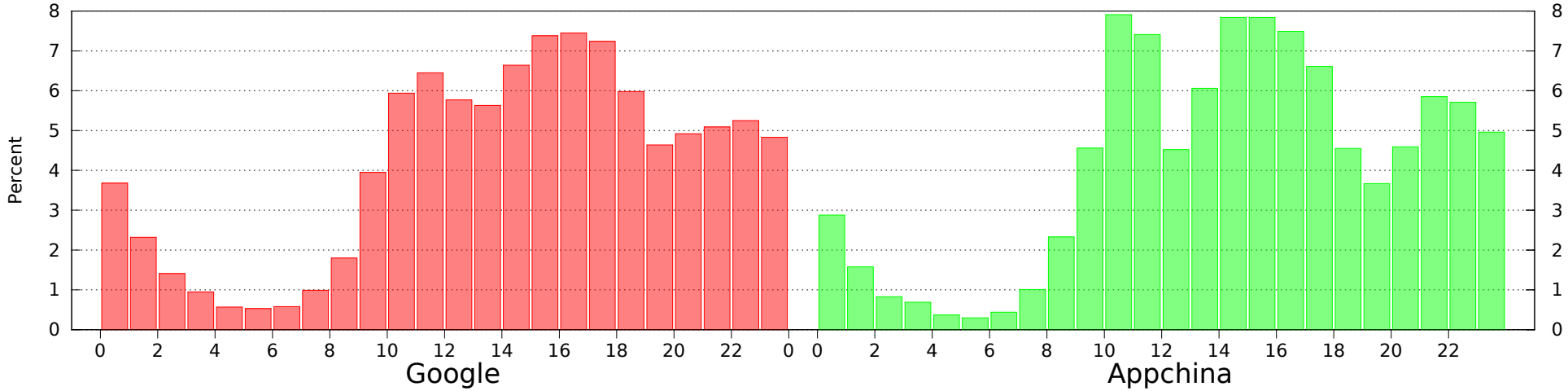
securityandtrust.lu

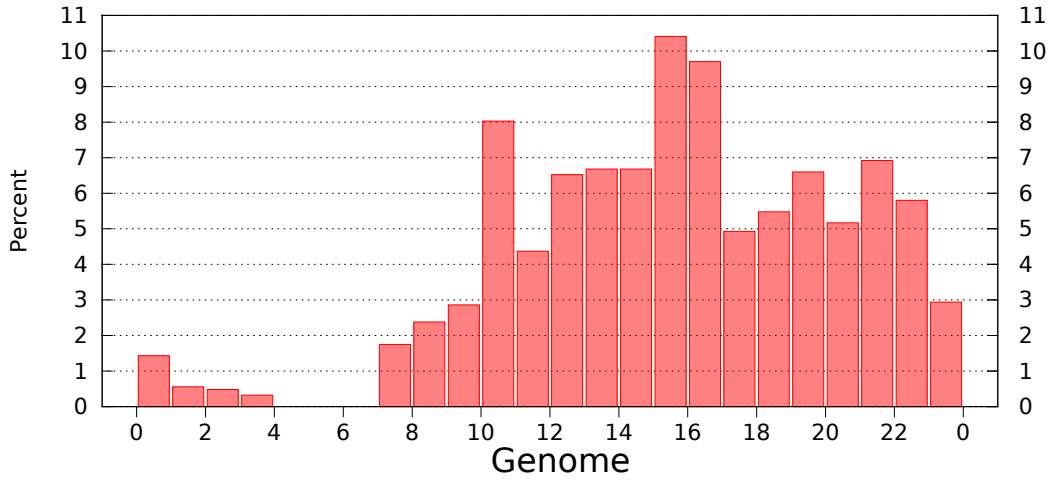
Distribution over each day of the week for each market

Market	Mon	Tue	Wed	Thu	Fri	Sat	Sun
Google	14.58%	15.85%	16.67%	16.32%	16.23%	10.32%	10.04%
Appchina	15.68%	14.35%	15.99%	18.03%	17.99%	8.24%	9.73%
Slideme	13.67%	15.57%	15.15%	14.81%	16.66%	12.28%	11.86%
Freewarelovers	14.42%	15.71%	15.88%	17.31%	14.30%	10.59%	11.80%
Proandroid	15.23%	17.17%	16.98%	15.19%	15.15%	9.84%	10.44%
torrents	15.76%	14.92%	15.53%	15.4%	15.30%	11.76%	11.49%
genome	14.31%	24.72%	18.84%	16.45%	13.75%	5.48%	6.44%

Reading: From all applications that were fetched from the Google market, 7565 (14.58%) were packaged a monday.

Distribution over hours





Distribution of applications over hours for three markets

Certificates

Certificate Reuse

Table: Number of Apps and certificates for each market

Market name	Apps	Cert Number	Apps per cert
Google	51,885	17,028	3
AppChina	63,476	13,193	4.8
Slideme	15,409	5,758	2.7
FreewareLovers	4,119	1,367	3
ProAndroid	2,633	1,371	1.9
Torrents	5,255	1,700	3.1
Genome	1,258	134	9.4
Total (dedup)	137,105	31,226	4.4

- Malware authors are more likely to reuse certificates

Interlude

securityandtrust.lu

The Question. . .

- We collected a grand total of 31,226 unique Certificates
- Can you guess how many of those are **not** self-signed?

Interlude

securityandtrust.lu

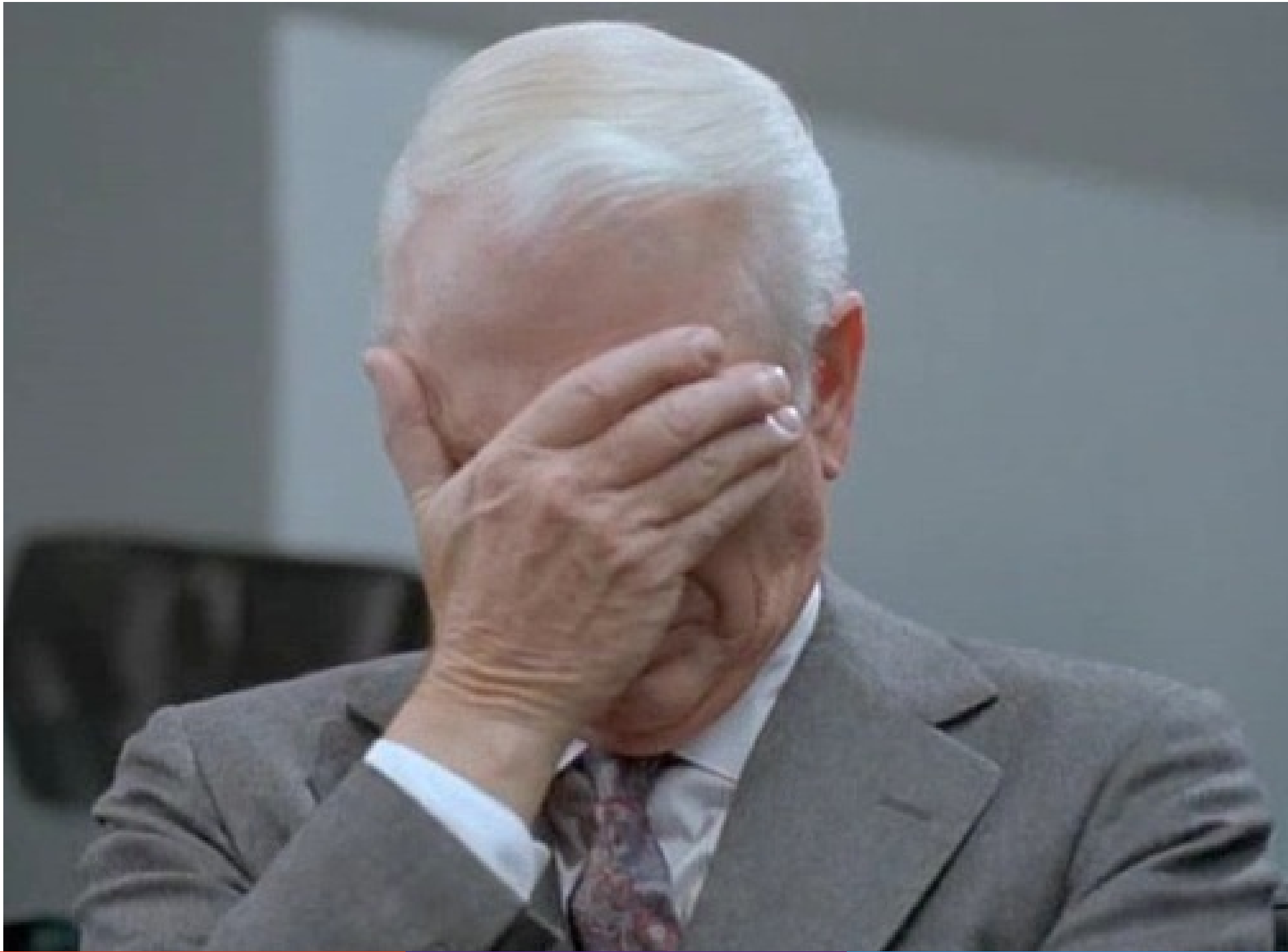
The Question. . .

- We collected a grand total of 31,226 unique Certificates
- Can you guess how many of those are **not** self-signed?
- Hint: Be pessimistic

Interlude

The Question. . .

- We collected a grand total of 31,226 unique Certificates
- Can you guess how many of those are **not** self-signed?
- Hint: Be pessimistic
- **31**



But then. . . What are those certificates actually certifying?

securityandtrust.lu

Not any kind of Identification. . .

- More precisely, not any kind of *trustable* Identification

Same Origin

- Nonetheless, a self-signed certificate allows to prove that two apps have the same origin
- Same-Origin actually useful on Android (Update, Inter-App communication)

A selection of top Malware certificates...

securityandtrust.lu

Apps#	Malware#	Certificate Issuer & Owner
3266	196	EMAILADDRESS=android@android.com, CN=Android, OU=Android, O=Android, L=Mountain View, ST=California, C=US
167	167	C=keji0003
281	165	CN=PhoneSniper, OU=Phone, O=Phone, L=china, ST=shenzhen, C=cn
98	98	CN=kejikeji, OU=kejikeji, O=kejikeji, L=kejikeji, ST=kejikeji, C=kejikeji
102	95	OU=Google Inc., C=US
52	52	CN=Fujian Kaimo Network Tech
30	30	CN=a, OU=a, O=a, L=a, ST=a, C=a
24	21	CN=Sexy
19	19	C=0
12	12	CN=lzq, OU=lzq, O=kdsjfkl, L=dlkfjkl, ST=fwekfj, C=430034

Who cares anyway?

securityandtrust.lu

Developers don't...

- Most of them don't bother to put relevant data into their certificates
- And when it *looks* relevant, it's probably copy/pasted directly from the first tutorial of a Google search
- Certificate is just the thing Eclipse is complaining about...

Users don't neither

- They don't know what a certificate is
- Even if they knew, they're never presented any certificate-related information anyway...

DOUBLE FACEPALM

FOR WHEN ONE FACEPALM DOESN'T CUT IT

Further experiments

securityandtrust.lu

Question

Is it possible to use metadata to improve our chances to find malware ?

Why ?

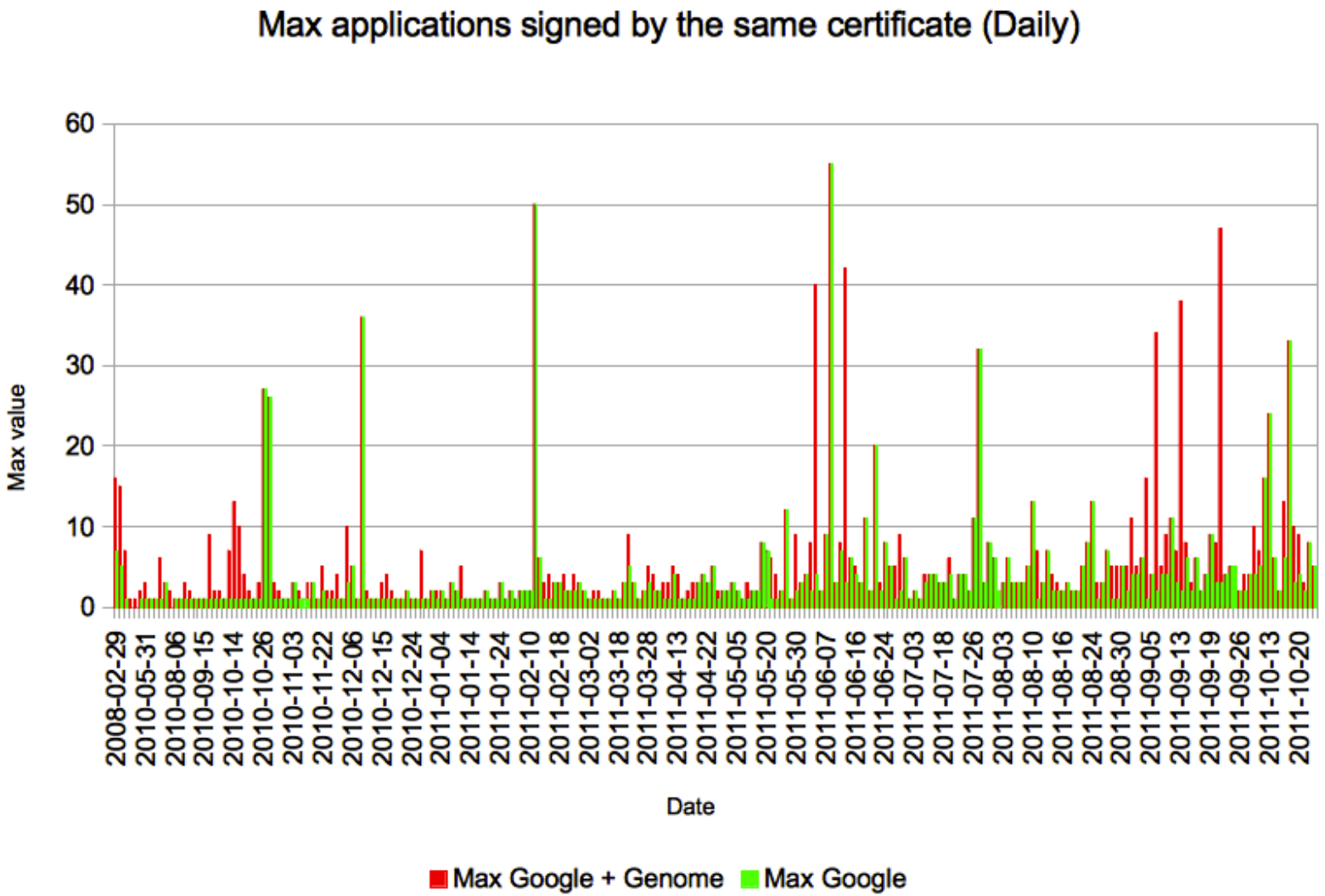
It could be interesting for malware analysis. Dealing with a reduced set of applications where the density of malware is higher than a random picking approach would be an asset.

Which metric to observe ?

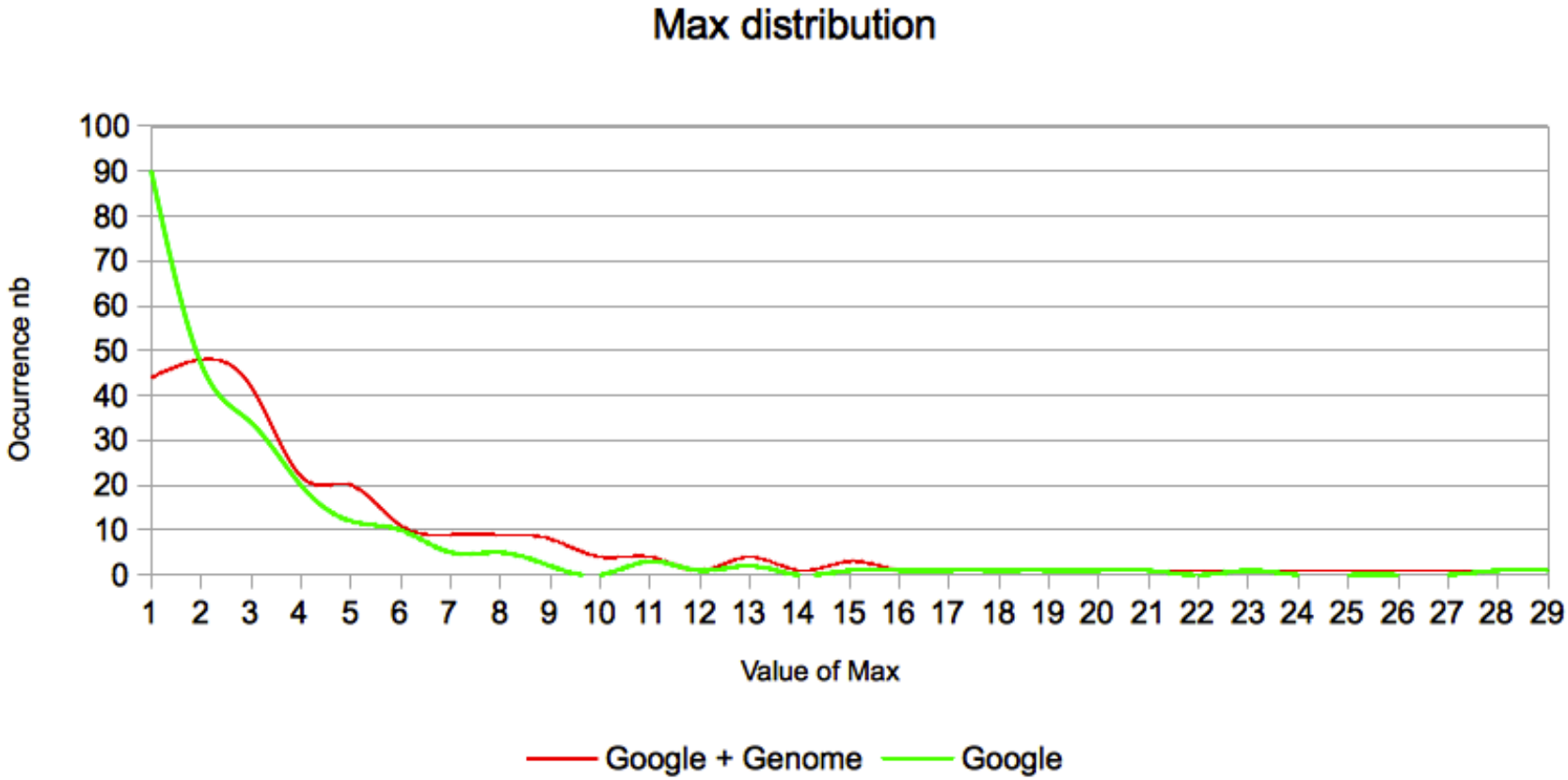
- We choose to screen certificates that signed more apps than a threshold per day
- Max = the cert. that signed the most of application in a given day

Graph 1

securityandtrust.lu



Graph 2



Note

If there is still malware in the market we would be able to catch some of them based on this metric !!!

- For identifying suspicious certificates we have to take those that have signed between 3 and 11 apps by day.

Experiments

securityandtrust.lu

Description

- Retrieve in the market all certificates who signed a given number of applications on a daily basis
- Get applications signed by each certificates on those days
- Get the proportion of malicious apps in this set

Parameters

- We chose to take 2<Sig. per day<12
- Dr. Web trial edition is used to detect the proportion of malicious apps

Results

securityandtrust.lu

Results

- We isolated 6,186 applications on over 52,000 apps
- In this subset 1,162 (i.e. 18.78%) are considered as malicious according to Dr. Web

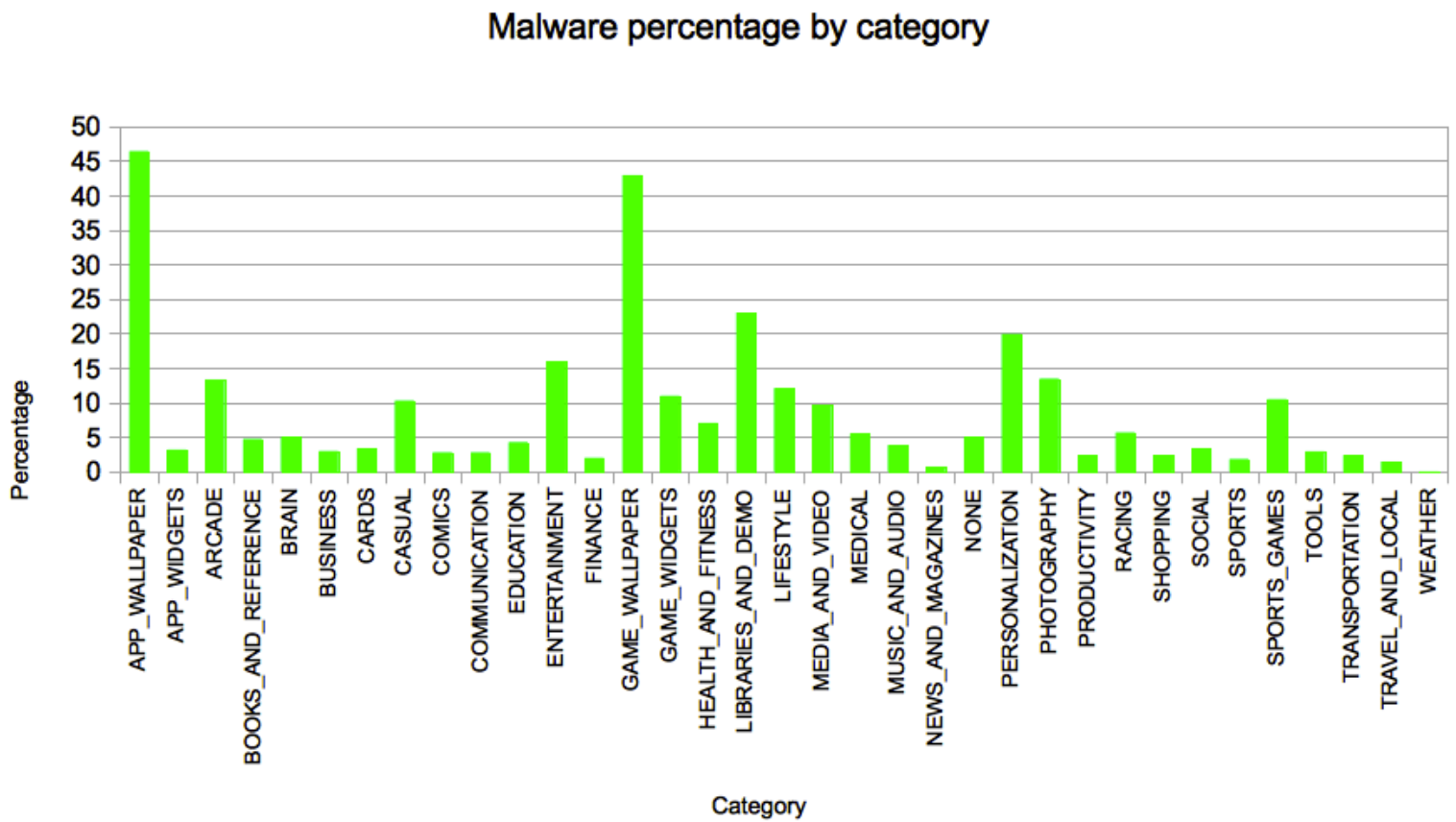
Some stats on this subset

- 678 cert.
- 6,186 applications -> avg=9.12 apps/cert
- 678 certs. have signed 15,914 distinct apps in Google play
- 678 certs. have signed 29,881 in the whole dataset

But, how many we would catch by picking apps randomly in the market ?

According to Dr. Web

- About 10.0% of applications are malicious (Surprised ?)
- Most of the malware detected are adware, spyware



Question

We have what we are looking for by picking in market categories

Yes but in our approach we did not take yet the information of malware distribution in different categories
Let's see if we can rely on packaging pattern for isolating malware in categories

Example : The APP_WALLPAPER category

By observing the same metric (i.e. certs. which have signed the most application on a given day)

- We isolate 334 apps for Sig. by cert. > 2
- 272 (81.44%) are 'malicious'
- 272=42% of malware in this category

Note

This search works also inside categories. We can increase our probability of finding malware.

Partial conclusion

securityandtrust.lu

It does not seems to scale to build a detection mechanism

- This metric does not seem to generalize enough malware.

We can use this to isolate a subset where we have more chances to find 'malicious' applications

Using the packaged apps by certificate metric we were able to isolate :

- A set where the probability to find malware is almost 2 times higher than a random selection in the market
- This technique can be used as a first step to find malware

By analyzing data we can observe strange 'programming' patterns ...

The best android programmer ever seen

securityandtrust.lu

The first and second place

The 29th of february, 2008 between 10am and 11am

EMAILADDRESS=android@android.com, CN=Android, OU=Android, O=Android, L=Mountain View, ST=California, C=US signed 615 apps

- sha1 : 61ED377E85D386A8DFEE6B864BD85B0BFAA5AF81

Damn it seems to be leaked certificate !

The same day between 8am and 9am

The same certificate signed 369 apps. Very tough morning for a developer.

The 25th of May, 2012 between 3pm and 4pm

CN=Emin KURA, OU=Development, O=TRISTIT, L=Istanbul, ST=Bahcesehir, C=TR signed 229 apps

- sha1 : 952F56BCA55EE75B64CBF0B245E9E6D334D45C42

The prize of the country that developed the most malware is for ...

securityandtrust.lu

This ranking is built from applications identified as malware by Dr. Web

Country	Nb of malware	Distinct Cert.	App. by cert
USA	765	116	6.6
China	303	50	6.1
India	297	60	4.95
Italy	190	19	10
UK	127	14	9.1
Russia	88	12	7.3
France	64	7	9.1

And the prize for most motivated malware developer is for ...

Italy

The mystery of leaked certificate

Wanted : EMAILADDRESS=android@android.com, CN=Android, OU=Android, O=Android, L=Mountain View, ST=California, C=US

Why a malicious author would like to use such a certificate ?

- Because they can use applications signed by the same certificate without asking permissions. Can lead to Application level privilege escalation !

Why a non rogue developer would like to use such a certificate ?

We do not know ...

- Application like : com.custom.lwp.sexybooty3.apk have been signed by such a certificate and it does not seem to be a Google application
- 28 apps signed by this cert. on our snapshot of Google Play (not detected as malicious)

Desktop malware VS mobile OS malware

securityandtrust.lu

The term of 'android malware' does not seem to be well defined

What is really an android malware ?

- Apps that leaks iMEI, IMSI are malware ?
  A lot of advertisement libraries do it !
- Are apps that propose you other applications are malware ? Which kind of apps are proposed, malware/goodware ...

Even AV cannot give us the answer

For a set of 2,000 apps randomly picked in the market

- Avast reported 11 threats
- Against 207 for Dr. Web

Avast seems to report only critical threats like Plankton malware while Dr. Web is focused on adware, spyware as well.

Open questions

securityandtrust.lu

Why a 'normal' developer would like to package a lot of applications in a short time ?

- Because he wants to spread malware ?
- He uses shared code ?

Is the certificate system put in place is efficient ?

- No trust chain
- Does a self signed certificate makes sense ?

The data shows us that the link between a true identity and a certificate is broken due to the self signed mechanism. Thus the only thing that we are pretty sure is that a cert. which have signed many applications corresponds to a unique developer. Unless it was leaked ...

SnT
securityandtrust.lu

Thank You!

Questions?