asurion

The Technology Protection Company

Optimizing Protection Systems for Emerging Mobile Threats

October 2012

INTRODUCTION

Mobile technologies and device sales are rapidly growing. In the United States alone, more than 60% of mobile devices purchased will be smartphones by 2013. Mobile application downloads for these devices will grow 72% to over 34 billion downloads.[1]

As smartphone penetration is targeted to surpass current computer market penetration in the US by 2017, malware is also forecasted to become as prevalent an issue on the smartphone as it is on personal computers. Industry studies predict that mobile malware will grow by up to 200% by the end of 2014.[1]

Several emerging trends have made the risk of malware threats even greater. Near field communication technologies (NFC) and mobile payments have increased the value which can be extracted from unsecure devices and transactions. In addition, synchronizing of information to personal computers and the increased use of mobile devices for transaction authentication has exposed an even larger set of consumer information to vulnerability. Additionally, with more companies experimenting with and deploying “bring your own device” (BYOD) policies, these security threats are growing larger for enterprises.

Unfortunately, two other trends have made isolation and eradication of malware significantly more challenging. First, malware developers are creating improved technologies and systems to avoid detection and exploit user information. At the same time, developers are being forced to find new business models in order to legitimately monetize their applications through techniques such as ad insertion and tracking/selling user data. Given these trends, malware detection systems need strategies to be even more flexible and accurate

Ten Key Strategies in Optimizing Protection Systems for Emerging Mobile Threats

Design

1. **Choose a Cloud-Based Architecture**
2. **Select a Massively Parallel Programming Model**

Intake

3. **Create Independent Intake and Analysis**
4. **Utilize Global Crawling Solutions and Acquire Paid Applications**

Analysis

5. **Develop a Rules-Based Approach**
6. **Add Context to Improve Resolution**
7. **Leverage Key Static Analysis Techniques**
8. **Collaborate for Parallel Cloud Anti-Virus Analysis**
9. **Leverage Dynamic Analysis Techniques**
10. **Apply Machine Learning**

[1] Frost & Sullivan - Mobile Cybersecurity Overview: A Brief Analysis of Trends in Q3 2012

when differentiating between malware and legitimate applications.

To address these growing threats, Asurion recommends ten key strategies and technologies. While these technologies are not to be considered a comprehensive plan to remediate all mobile malware, the benefits will help wireless carriers and enterprises identify, analyze and stop malicious software attacks more rapidly and minimize chances of exploitation and loss.

Design – Building for Speed and Flexibility

Having a clear view of key objectives before designing a complete malware protection system is critically important. Within this type of system, the two most critical objectives should be: 1) speed to quickly intake and analyze samples and 2) flexibility to observe samples and behaviors from multiple perspectives. With advances in systems technology, a cloud-based system utilizing a massively parallel programming model not only satisfies these objectives, it can establish a platform that generates insights on additional application features such as privacy usage, battery consumption, and network utilization.

Choose a Cloud-Based Architecture

While it may seem obvious, protection systems must be designed with cloud-based architecture principles in mind. The system must have the flexibility to use hundreds or even thousands of spot processing instances on demand, at low marginal cost, or the system will not have the computational underpinnings required. This not only includes processing required for critical large-scale threat analysis, but also continuous real-time services and unsupervised machine learning.

That said, in order to leverage the cloud for mobile security, additional critical elements must be considered. For example, a design that is based upon a classic n-tier architecture that happens to run within a cloud would simply be cloud hosting and not be sufficiently flexible. In addition, a NoSQL-based design would not provide enough parallelism to rapidly or sufficiently analyze samples.

Unless the cloud architecture is designed to use a massively parallel programming model, the analytics of the system will be severely handicapped.

Select a Massively Parallel Programming Model

When malware rules are added to detect the latest and most advanced malware, each app must be re-examined and updated with its new analysis results. With many thousands of apps to re-analyze, this process can take days or weeks to complete. Massively parallel programming models solve the problem so that the processing time can be reduced to days, hours, or even minutes.

For example, one such programming model known as MapReduce enables the processing of complex problems across vast datasets using a large number of spot instances in the cloud. The process consists of a mapping step and a reduction step.

In the mapping step, a master node divides a large problem into smaller, simpler problems, which are then distributed to worker nodes. Worker nodes further divide and distribute the problem to other worker nodes until the simplest problem can be solved directly by a worker.

In the reduction step, the master node collects and combines the sub-answers to solve the original problem.

By leveraging massive parallelism, or with similar programming processing, time can be dramatically reduced. This programming model also has the benefit of being optimized to support classic rules-based analysis and large-scale analytics used by machine learning systems for anomaly detection, behavioral analysis, clustering, unsupervised machine learning, data mining, and other forms that can produce unique insights from such big data.

INTAKE - CREATING RAPID SAMPLE INTAKE FOR QUICK REMEDIATION

When designing a malware detection system optimized to detect and quickly remove malware from a mobile population, one must not overlook its most essential component: The solution must find, analyze and determine how to remediate large numbers of malware samples as quickly as possible from the mobile device population and application markets.

Create Independent Intake and Analysis Processes

Intake is best decoupled from analytics so that samples may come from a variety of sources, including devices within the mobile population, app markets, fake markets, black markets, mobile backup systems, website submissions, malware exchange consortia, and so on.

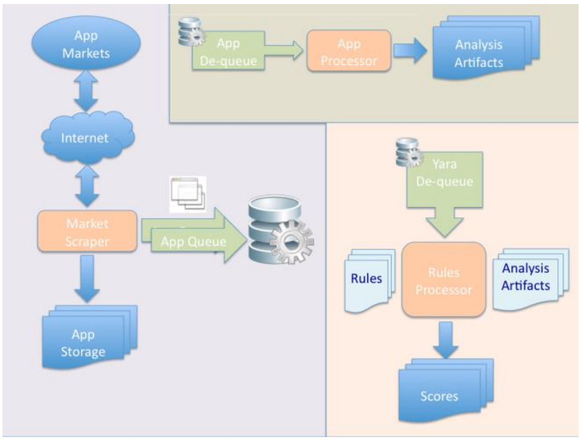


Fig 1: Sample of separate intake and analysis flows

It is this rapid intake, critical for core analytic capability, which will distinguish the effectiveness of one security cloud from another.

Utilize Global Crawling Solutions and Acquire Paid Applications

Specialized application “web crawlers” should be built or licensed to help with intake by downloading free apps from official US marketplaces (e.g., Google Play, Apple iTunes Store, BlackBerry App World, and Windows Phone Marketplace) as well as international marketplaces (e.g., 1mobile, nashymarket.ru, and Docomo).

Unofficial markets in Russia and China (there are dozens) are where most of the mobile malware activity is found. Underground markets and fake markets also serve as major distribution points for malware.

Additionally, to fully protect a respective user-base, all paid applications across high-risk categories should also be acquired and the focus of analysis.

ANALYSIS – MAXIMIZING MALWARE IDENTIFICATION ACCURACY

Once a sufficient computational platform is designed and an intake process is established, a leading malware prevention system should utilize five key elements to establish high accuracy in its malware identification. Specifically these are:

- Develop a rules-based analysis approach
- Leverage key static analysis techniques
- Partner with leading industry players to create a parallel cloud AV analysis process
- Leverage additional dynamic analysis techniques
- Utilize supervised and unsupervised machine learning

Develop a Rules-Based Approach

Analytics in the security space have traditionally meant a rules-engine model applied to the traits and behaviors extracted from an application.

For example, if an application is observed logging and transmitting private data to a known malicious source, one can make a rule that will trigger in that situation.

In this model, an application is decomposed into its behaviors, and then a rules engine is used to determine whether some of its behaviors, taken together, indicate malicious activity.

Apps that violate these deterministic rules are then flagged, quarantined or deleted depending on how the security provider curates malware within the population.

A key benefit of using this rules based model is that rules based on malware families tend to be very accurate because they exhibit unique fuzzy patterns designed to match the family. Because of this, until new applications in the family change their patterns enough to not match those previously identified, the rule(s) will catch large amounts malware.

Leverage Key Static Analysis Techniques

When analyzing applications, the first and most common method for identifying malware has been through static analysis techniques. These techniques study application code and properties to see what the code may do, usually without ever installing the application. Static analysis can be useful for studying all aspects of the application, including code that may not run right away or that may be triggered by a user action that we may not encounter by just installing or running the app.

The most effective static analysis techniques can be grouped into three categories of signatures, events, and process/flow analysis.

Signatures

One of the most popular forms of signatures is hash-based. Excluding polymorphic and metamorphic malware—which are still virtually unknown in mobile malware today—previously identified malicious samples are easily recognized by comparing the cryptographic hash of the unknown sample to hashes of previously observed malware. If the two hashes match, the sample is previously known malware and the system may then reject malicious samples.

Another effective signature to analyze is the developer's signature. While it may seem obvious, many developers of malware can often

Basic Rules Creation with YARA

YARA is an open-source, rules-based engine aimed at helping malware researchers identify and classify malware samples. With YARA, one can create descriptions of malware families based on textual or binary patterns contained in samples of those families. Each description consists of a set of strings and a Boolean expression which determine its logic.

```
rule silent_banker : banker
{
  meta:
       description = "An example"
       thread_level = 3
       in_the_wild = true

    strings:
       $a = {6A 40 68 00 30 00 }
       $b = {8D 4D B0 2B C1 83 }
       $c = "UVODFRYSIHLNWPEJXQZAKCBGMT"

    condition:
       $a or $b or $c
}
```

The rule above tells YARA that any file containing one of the three strings has been identified as the malware known as silent_banker.

This is just a simple example. More complex and powerful rules can be created by using binary strings with wild-cards, case-insensitive text strings, special operators, regular expressions and many other features found in YARA's documentation.

be identified through a relatively simple set of analyses. Once a questionable developer is identified as the creator or contributor to an application, the sample can similarly be quarantined or rejected.

Events and Interactions

In addition to signatures, certain events are also commonly used to identify malware. For example, an application communicating with a known malicious server should quickly be isolated and brought to the attention of an analyst.

That said, a majority of malware cannot be identified by one specific event but rather by a series of events in a specific order. For example, an app that exfiltrates the device ID to a remote server without the user's explicit consent is completely normal behavior for an ad-supported app.

Because the diversity of free software has now evolved to share many of the same characteristics of classic malware, they are often indistinguishable without much higher resolution scanning provided by a solution such as Call-Flow Graph (CFG) analysis.

Add Context to Improve Resolution

CFG scanning statically traces execution paths through the program so that one can determine whether certain API calls are used in the context of a chain of calls.

For example, a malicious application may only be distinguished from other legitimate applications solely by a lack of user interaction prior to an on-device purchase. Such call-flow context is very difficult to achieve without the higher resolution scanning provided by CFG analysis systems. Asurion plans to focus on this pioneering technology in a future white paper.

In the end, the success of static analysis process depends heavily on the traits that one can extract from the sample, and how well those traits or behaviors are selected to differentiate malware from legitimate software.

Collaborate for Parallel Cloud Anti-Virus Analysis

In order to identify malware that is known by the malware community in general, a technique known as collaboration utilizing “Cloud Anti-Virus” or “Cloud AV” should be used.

As noted by AV company Symantec:

> “While Symantec created nearly 1.8 million new virus definitions in 2008, the reality is that the signature approach and other traditional methods of security are not keeping pace with the number of threats being created by online criminals.”

As the number of PC malware samples has exceeded the ability of individual anti-virus companies to keep pace, they have joined forces to exchange samples, and help one another battle malware.

When multiple companies collaborate to run anti-virus engines in parallel, studies have shown improved overall detection rates—up to 98% in one study—without significant increase in false positives as samples missed by one company may be caught by several others.

This parallel scanning process has now extended from PC anti-virus detection to mobile virus detection as well. One may privately contract with a number of anti-virus companies to scan samples via commercially available scanning engines arranged in parallel. This approach then detects malicious samples across a spectrum of companies and participating consortia, and to some extent *within* the anti-virus community.

Cloud-AV essentially sets the floor for what may be detected. However, much more must be built on that foundation to create a useable cloud security system.

Leverage Dynamic Analysis Techniques

Utilizing the above static analysis and cloud AV analysis techniques, basic systems can begin identifying a large portion of more easily identifiable malware. One problem with static analysis, however, is that developers are now becoming more skilled at arranging code so that it’s difficult to tell what the code may do. Some portions of the code may be encrypted so that they are not subject to analysis until the application runs. Only after running do these types of programs decrypt the hidden portions and begin executing malicious code.

The most accurate malware identification systems will have both on-device and cloud based analysis elements to best observe and analyze applications’ dynamic behavior.

On a server, applications can be installed on a virtual instance of a mobile device. Once running, the application can be safely observed in a quarantined environment to identify any malicious or unexpected behavior.

On the client side, applications such as Asurion’s AppAsisst and Google’s Bouncer profile the behavior of applications on physical devices running in real world conditions. Next, using additional cloud-based processing, the application can be further analyzed from multiple perspectives—such as privacy, security/best practices, battery conservation, and network utilization—as well as analyzed across multiple platforms.

Not only does this client-based and cloud-based approach help massively increase malware identification and removal, but the solution also provides a research platform which can then support multiple simultaneous experiments that are pluggable and served generically through a common infrastructure. With this infrastructure, unique questions and experiments can be executed seamlessly on a large scale.

Apply Machine Learning Techniques

Once static, cloud AV and dynamic analysis techniques have begun to identify malware at a standard speed and accuracy, supervised and unsupervised machine learning should be implemented to assist with speed of identification as well as application trait extraction and differentiation.

Supervised Machine Learning

Machine learning is most useful when analysis, such as malware analysis, is heavily dependent upon clustering, or analyzing the distances between key elements.

To better understand clustering, imagine you are analyzing fruit, and the feature you are studying is color. As you analyze the fruit, you are plotting the resulting dots onto this strip of paper next to their color.

The samples that cluster together are similar to each other, and the ones that are far apart, will be different.

If you want to go further and distinguish between bananas and lemons, you will need to throw in more traits, or dimensions, that you are plotting. The paper will take on the shape of a two-dimensional, then three-dimensional grid.

Using this technique to analyze malware can provide a similar multi-dimensional analysis on applications samples. Plotting a few samples of known malware on a graph allows analysts to notice that the malicious programs tend to land in a cluster (i.e., a red cluster) and legitimate programs land in a different cluster (i.e., a blue cluster), such as in the graph shown below.

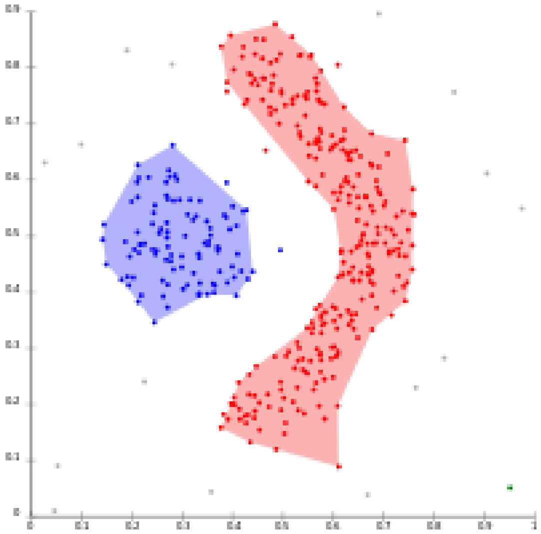


Fig 2: Malware clustering example

If an analyst were to plot an unknown sample and discover that it lands in the red area, she may reasonably guess that the sample is malware. In addition, an analyst monitoring image files may also notice a sample that falls far outside the expected cluster, which may indicate that it's not an image file at all. It may be a native library attempting to escape detection. Clustering can reveal potential threats in both clusters and outliers.

As data and relationships become more complex, machine learning systems have the unique power to plot these complex clusters in N-dimensional space, beyond what most humans can visualize. Further tools and optimizations—such as higher-dimensional PCA K-Means, different kinds of Bayesian analysis and Gaussian processes—may then be implemented.

Eventually, as a company develops a competency powerful machine learning, it will not only dramatically increase the capability of applying generic anomaly, classification, and prediction analysis to malware detection, but also across different types of corporate functions.

Unsupervised Machine Learning

Once a rules-based engine and basic machine learning have been implemented, unsupervised machine learning can further accelerate the pace of malware detection.

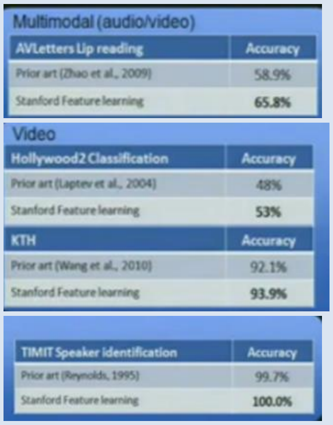
In supervised machine learning, systems are instructed by a researcher to look for specific traits. In unsupervised machine learning, a system discovers its own set of traits and determines which of these traits are meaningful and significant.

Utilizing such a powerful solution in malware detection is essential to have within any advanced security cloud, helping identify misbehavior as well as other qualities of applications ranging from privacy, security, battery life, network utilization, and usability. Asurion has plans for further white papers focused on leveraging insights from this exciting new field.

Rapid Advancements in Unsupervised Machine Learning

While unsupervised machine learning is still a pioneering field of research, the advances gained by this screening method should not be underestimated. Specialists in machine learning at Stanford University have now leveraged machine learning to significantly exceed the analytical capabilities of domain experts in fields such audio processing, video, and object classification without knowing anything about these subjects.

To appreciate the gravity of these statistics, realize that domain experts discovering a 0.1% improvement against established benchmarks in their field is extremely significant. Now, with unsupervised machine learning, researchers knowing nothing about a particular field have been leapfrogging specialists by 1.2% which represents an advancement of 120%, or nearly a decade of incremental improvements made by domain experts over the years.



Multimodal (audio/video)

AVLetters Lip reading	Accuracy
Prior art (Zhao et al., 2009)	58.9%
Stanford Feature learning	**65.8%**

Video

Hollywood2 Classification	Accuracy
Prior art (Laptev et al., 2004)	48%
Stanford Feature learning	**53%**

KTH	Accuracy
Prior art (Wang et al., 2010)	92.1%
Stanford Feature learning	**93.9%**

TIMIT Speaker identification	Accuracy
Prior art (Reynolds, 1995)	99.7%
Stanford Feature learning	**100.0%**

Fig 3: Stanford research results

In Conclusion

When implementing a mobile malware protection solution, thoughtful design planning, intake strategy and analytical capabilities is important. In doing so, the security cloud that you design today will not only be able to gracefully evolve into an extremely powerful malware detection and remediation system, but can also be the foundation for big data opportunities afforded by your company's position in the mobile space.

This paper was designed as a brief primer to describe key factors and technologies to be considered in creating a leading mobile malware protection system. However, the mobile security industry and Asurion research are rapidly evolving and producing new insights on emerging malware threats and solutions. Should you wish to learn more about key topics such cloud infrastructure design, high-accuracy malware identification or other mobile security research trends, please contact our Asurion Research & Development team at Christopher.Reynaga@asurion.com.

asurion

Will Your Solution Keep Up with Emerging Mobile Threats?

Mobile threats are evolving quickly. In order to keep up with this rapid growth and minimize exposure, ensure your systems are employing the following key elements in their design, intake and analysis:

□ **A cloud-based system**
Is your system designed to have the flexibility to use thousands of spot processing instances on demand required for large-scale malware analysis and continuous real-time services?

□ **Built on a massively parallel programming model**
Can your programming model support large-scale re-examination and updating of hundreds of thousands of apps in a short amount of time by capitalizing on a distributed cloud architecture?

□ **Maintaining independent intake and analysis processes**
Is your system's intake design and technology constructed to maximize the speed which it can find and process samples from a variety of sources, including devices, app markets, fake markets, black markets, mobile backup systems, websites, malware exchange consortia and more?

□ **Utilizing global crawling solutions and acquiring paid applications**
Will your system quickly and automatically find the broadest array of free and paid applications, not only from official U.S. and international marketplaces but also unofficial markets such as those in Russia and China where most mobile malware activity is found?

□ **Leveraging a rules-based approach**
Does your system identify families of malware to catch large groups of apps en masse by using a rules based approach?

□ **Adding context to its approach to improve analytical resolution**
Can your system determine when API calls are legitimate or illegitimate through the use of high-resolution scanning capabilities which trace execution paths through the programs?

□ **Employing key static analysis techniques**
Can your system quickly identify characteristics of applications without running the application, by locating known signatures, or patterns of events which identify a threat to the user or the network?

□ **Sharing and incorporating results in parallel cloud anti-virus analysis forums**

Is your system's accuracy further improved through the sharing and incorporating new rules and signatures with other industry partners?

□ **Leveraging dynamic analysis techniques**

Will your system identify malware that may have dormant, encrypted sections of code which elude static analysis techniques and only inflict harm when the application is operational?

□ **Applying supervised and unsupervised machine learning**

Can your system rapidly produce new insights through clustering which identify behaviors and traits of harmful applications? Will it utilize unsupervised machine learning to accelerate the pace at which new rules are created and malware is detected?

Should you wish to learn more about key topics such cloud infrastructure design, high-accuracy malware identification or other mobile security research trends please contact our Asurion Research & Development team at Christopher.Reynaga@asurion.com.