ENTERPRISE MOBILE THREAT RESEARCH

HospitalGown: The Backend Exposure Putting Enterprise Data at Risk

Q-2 2017

appthority
MOBILE THREAT TEAM

Table of Contents

Executive Summary

Securing an enterprise in the digital age means securing its data - protecting against the insecure handling of data at rest and in motion. And, since so much of a company's data is now accessed by mobile devices and apps, enterprises need to have a firm understanding of the risks to enterprise data stemming from mobile use.

This report shares research about a newly discovered data exposure vulnerability dubbed "HospitalGown" that highlights the connection between mobile apps and insecure backend databases containing enterprise data. HospitalGown is a vulnerability to data exposure, caused, not by code in the app, but by the app developers' failure to properly secure the backend servers with firewalls and authentication.

As our findings show, weakly secured backends in apps used by employees, partners, and customers create a range of security risks including extensive data leaks of personally identifiable information (PII) and other sensitive data. They also significantly increase the risk of spear phishing, brute force login, social engineering, data ransom, and other attacks. And, HospitalGown makes data access and exfiltration far easier than other types of attacks.

The Appthority Mobile Threat Team (MTT) discovered HospitalGown using an innovative technique that combines Appthority's market-leading dynamic app analysis with a new back-end scanning method specifically designed for discovering insecure (mobile) back-end servers. We used this technique to analyze the network traffic of over a million enterprise mobile iOS and Android apps for the HospitalGown vulnerability.

Key findings from the discovery include:

- The enterprise threat is real:
  - The apps connect to unsecured databases on a range of popular enterprise services, including Elasticsearch and Amazon Web Services
  - Enterprise security teams do not have visibility into the risk due to the risk's location in the mobile app vendor's architecture stack
  - In multiple cases, data has already been accessed by unauthorized individuals and ransomed
- The data exposure risk is significant:
  - Appthority found 1,000 affected apps on enterprise mobile devices connected to over 21,000 open Elasticsearch servers, revealing almost 43 TB of exposed data
  - A subset of just 4% of the affected apps revealed that as much as 163.53 GB of data, or approximately 280 million records, have been exposed
  - Data being leaked contains Personally Identifiable Information (PII) including: passwords, location, travel and payment details, corporate profile data (including employees' VPN PIN reset tokens, emails, phone numbers), and retail customer data

Apps with the HospitalGown vulnerability pose a direct risk to enterprises. In addition to the initial loss of data via the unsecure backend, is the risk of a secondary and direct attack on the primary enterprise data stores using data acquired via the initial breach. HospitalGown opens the mobile app clients to an easy breach, exfiltration of sensitive data, and the costs from remediation, lawsuits, compliance infractions and loss of brand trust. And, even if an affected app is removed from devices or the app store, the data collected is still available in an unsecured data store and at risk of being copied or downloaded by unauthorized parties.

1k

Affected Apps

21k

Open Servers

43TB

Exposed Data

Every new mobile app that uses a back-end platform for data storage or analysis is a potential source of risk. Enterprises relying on software developers to properly code and configure the backend connections are exposed.

Combined with the fact that in 2016 over 25% of organizations didn't detect breaches for a month or more[1], the ease of data access and the extent and nature of the data exposed, HospitalGown strongly underscores the need for enterprises to protect themselves with a comprehensive mobile threat protection solution that includes the new deep mobile application analysis.

[1] Quartz Media: https://qz.com/978601

Introduction

OVERVIEW

Data leaks are the submerged portion of the security risk iceberg, often unseen and unnoticed until reported by customers, law enforcement or worse, a hacker demanding a ransom to return access to the data. In fact, in 2016, over 25% of organizations took a month or more to notice a data breach incident and for one in eight, discovery took over a year.[2]

In 2016, over 25% of organizations took a month or more to notice a data breach incident and for one in eight, discovery took over a year.

With the recent media coverage over technically complex mobile threats like MilkyDoor and DressCode - threats that open tunnels through enterprise firewalls and potentially allow access to internal network resources to the outside world - it is easy to forget that massive data leaks and vulnerabilities that can cause them don't always go through the front door.

As a company whose mission it is to protect enterprise data, Appthority has long been a proponent of threat detection and education related to data risks. The Appthority Mobile Threat Team (MTT) is constantly innovating detection techniques to stay ahead of the risk curve and has recently developed a new detection and analysis technique which revealed a serious and significant data exposure in enterprise environments.

We've named this threat "HospitalGown" because of the backend data exposure risk it reveals. Apps with the HospitalGown vulnerability pose a direct risk to enterprises, opening them to an easy breach, exfiltration of sensitive data, and the costs from remediation, lawsuits, compliance infractions and loss of brand trust.

HOSPITALGOWN DEFINED

HospitalGown is a vulnerability to data exposure caused, not by any code in the app, but by the app developers' failure to properly secure the backend servers with which the app communicates. To enable rich functionality, developers often use backend servers to store persistent user data and programs like Elasticsearch to mine and analyze the data. Elasticsearch does not have built in security and access control and relies on external implementation of these security features with an authentication plugin or API for access, for example. If the Elasticsearch server is publicly accessible on the internet without these security features implemented, the data stored there will be available to anyone who knows where to look.

HospitalGown is a vulnerability to data exposure caused, not by any code in the app, but by the app developers' failure to properly secure the backend servers with which the app communicates.

HospitalGown, thus, constitutes a new class of mobile threat that is different in three key ways.

1. It exposes vast numbers of Big Data records that hackers can mine with minimal effort.
2. It is difficult for enterprises to detect because they almost certainly won't know when a breach of an app vendors' backend has occurred. As we confirmed, even the app vendors themselves didn't know. Unfortunately, without proper threat protection, the first sign of a breach may be a ransomware note demanding payment, or a highly targeted spear phishing attack campaign with the PII collected from massive mobile backend breaches.
3. Closing the vulnerability is outside of the enterprise's control. Only backend platform configuration improvements and possibly code changes within the affected app will eliminate the vulnerability. If the vulnerability is exclusively on the backend, even updating the app will not solve the problem.

2. Verizon 2017 Data Breach Investigations Report

METHODOLOGY

The Appthority Mobile Threat Team suspected that backend servers might be an important vector for data exposure and, in March 2017, began looking for threats to enterprises from poorly secured backends connected to apps in enterprise environments.

To explore this potential data risk, the MTT developed and used an innovative technique that combines Appthority's dynamic analysis of the app with a new back-end scanning method specifically designed for discovering insecure (mobile) back-end servers. The new technique now complements our dynamic mobile app analysis to determine whether a mobile app's communication with backend servers is secure. Initial efforts revealed data exposures on multiple backend platforms, including:

- Elasticsearch
- Redis
- MongoDB
- MySQL
- CouchDB
- CouchBase

The large volume of exposed data (43 TB) precluded a detailed analysis of every vulnerable backend service, so we focused on just one platform, Elasticsearch. We chose Elasticsearch because it is the back-end technology of choice for larger enterprise level data sets, often called "Big Data", and has been the target of recent ransom attacks.

The MTT then used two threat models to investigate exposed backend connections and apps. The first focused on scanning over the internet to identify unsecured servers with exposed data. In this model, although a secure mobile API routes traffic to an Elasticsearch cluster with stored user and enterprise data, the Elasticsearch servers are left unsecured. Using the information collected from apps during dynamic analysis, we were able to connect the dots back from some of these found servers to apps we had previously analyzed.

The second threat model examined the vulnerability from the the app direction. In this case, the mobile app interacts directly with an unsecured Elasticsearch server. In this worst-case scenario, a bad actor could:

- Reverse engineer a mobile app to obtain an Elasticsearch server IP address
- Scan the internet for servers with similar weak configurations or scan a company's network for additional insecure servers
- Listen for clear traffic being sent to the Elasticsearch server IP

Using this threat model, the MTT dynamically analyzed the network traffic of over a million iOS and Android enterprise mobile apps, looking for mobile apps that write directly to a backend server without appropriate security measures in-place.

These two threat models revealed tens of thousands of Elasticsearch servers that were exposing data and more than 1,000 apps affected by the HospitalGown vulnerability. We narrowed our focus for deeper research to 39 popular apps that were not adware and which had the most relevance to enterprises. The list of these apps can be seen in Appendix A.

Findings & Implications

FINDINGS OVERVIEW

In all, the MTT's analysis uncovered over 21,000 Elasticsearch servers that were exposing data and over 1,000 apps present in Appthority customer enterprises with the HospitalGown vulnerability. These apps were using weakly secured backend server infrastructure and leaking data at an astounding rate. Over 163 GB of data containing 280 million records were leaked by just the 39 apps we investigated more closely. (See Appendix for a list of these apps). These apps were confirmed to be exposing large amounts of data, including personally identifiable information (PII) and highly sensitive corporate data.

It is important to note that the risk of mobile apps using unsecured backend servers is not isolated to free and knock-off apps. Appthority discovered vulnerabilities in a wide range of enterprise and personal apps, some developed by respected vendors with well vetted security practices.

Vulnerable mobile apps were found in all of the following app categories:

- Agricultural operations
- Content management
- Dating
- Education and student tracking
- Enterprise mobile security and access management
- Games
- News
- Office productivity
- Travel, flight and hotel

While we limited our examination of the exposed data to less than one percent of the exposed data (see Appendix B for how data was handled), it was clear that other highly sensitive data was exposed. This data could be used to facilitate other attacks, conduct fraud or be sold to other bad actors.

The potential monetary and brand risk to enterprises is staggering if you consider the current breach cost of over $100 per record.[3] Even more sobering is the very real possibility of ransomware attacks that would not need to rely on a successful malware phishing attack for system penetration and could be deployed to any or all of the companies exposed. One of the apps for which we provide a case study in this report showed that the data had already been ransomed.

3. Ponemon Institute Data Breach Costs Report, June 2016

IMPLICATIONS

Unsecured Backend Data Stores: High Risk But Not New

The research uncovered examples of weakly or unsecured backend server infrastructure in use by apps that seemed to echo a report on unsecured Elasticsearch backends found on AWS written earlier in the year. These servers were accessible from the Internet, lacked any means of authentication to prevent unwanted access to the data they contained, and failed to secure transport of data, including PII, using HTTPS: conventions.

It is worth mentioning that this goes against documented Best Practices by Elastic, Redis, and others for app development using these types of data services. While the best practices report identified that the lack of access controls on openly accessible data stores left the servers' data vulnerable to ransomware, our research found the risk goes well beyond that.

Weakly secured backends leveraged by mobile app developers create opportunities for big data leaks and a significant increase in the risk of data misuse for spear phishing, brute force login, or other types of PII-based attacks for enterprises with employees, partners, or customers that use or have ever used these apps. 81% of hacking-related breaches in 2016 leveraged either stolen and/or weak passwords.[4] Addressing HospitalGown is a critical security measure to take to ensure your enterprise is not low hanging fruit for an easy hack or extensive breach using leaked data.

HospitalGown: An Easy Way to Access High Value Data

Hackers looking to steal data and commit ransomware attacks are changing tactics, focusing on vulnerabilities in the way mobile apps and devices access data. They are also now able to use big data techniques to correlate small bits of information collected via smaller attacks into a profitable black-market commodity or actionable intel. In fact, the threat of data leakage from mobile-sourced breaches is now feared to be greater than PC-based breaches.[5]

Traditionally, attacks aimed at acquiring access to data require multiple steps and almost always require a compromised user, network or device as a first step. From this landing zone an attacker can begin sniffing out small bits of sensitive information, unencrypted credentials, PII or leverage insufficient user authentication and authorizations to compromise enterprise networks and data stores. For example, the recent MilkyDoor Android malware requires a reverse SSH tunnel and proxy functionality in order to gain access to internal enterprise network resources. This type of attack can be expensive and time consuming to perpetrate with no guarantee of a pay day. The attack on a corporate network may come months after a device is actually compromised.

HospitalGown is based, not on malware, but on a vulnerability in the a mobile app's architecture and infrastructure, and uses techniques more commonly found in web site attacks. The attacker looks for vulnerabilities in the space between the vendor's mobile application and the app's server side components. The servers for most mobile applications are cloud based and accessible via the Internet, this allows a bad actor to skip the long and potentially many-layered "compromise" stage of an attack, accessing company data directly from a database that is impossible for the enterprise to see or secure.

4. Verizon 2017 Data Breach Investigations Report

5. The Growing Threat of Mobile Device Security Breaches, Dimension Research, April 2017

Data Exposure

DATA EXPOSURE EXAMPLES

To illustrate the risk of data loss from the mobile apps' insecure backend communications, the MTT documented the exposure from two diverse examples of enterprise apps.

Example 1: Pulse Workspace

Pulse Workspace, is an app by Pulse Secure LLC,a leading provider of secure access and mobile security solutions to enterprises, government agencies, and service providers. The Pulse

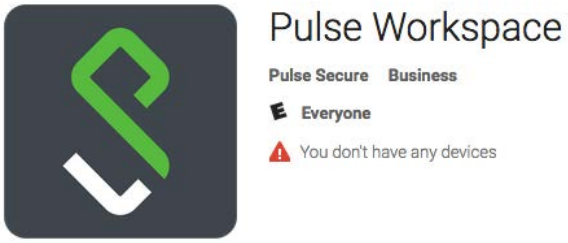


Figure 1

Unsecured Mobile Backend With API: Pulse Secure Case Study

Workspace app provides an environment to corporate employees for accessing enterprise network and web applications. Despite using an API to secure front-end Elasticsearch access, the backend, and all of the app's data records, were exposed (see Figure 1 below), leaking the data of Pulse's customers.

To get a sense of the data exposure, we assessed the first few hundred records from each of approximately 100 indices exposed on the Pulse Secure Elasticsearch server. This data sample amounted to 35 MB out of 7.97 GB of exposed data or less than 0.5% of the exposed data.

This limited view into the data revealed mobile app, alert, and VPN related user data for Pulse Secure and customer organizations including:

- A power management company from Charlotte, NC
- A Middle-East cyber security services, solutions and consultancy provider
- An IT service provider for hedge fund management companies
- A critical asset protection company
- A pharmaceutical and medical service provider
- A U.S. Federal Court
- A U.S. missile company
- A CCTV surveillance company
- A computer security and antivirus company
- One of the largest U.S. telecom carriers
- One of the largest privately-owned construction and civil engineering companies in the U.S.

Further, this data contained PII, such as:

- Full Employee Name
- Email addresses
- Phone numbers
- PIN reset tokens
- Passcode lengths
- Account dates
- Last seen dates
- Device Information (IMEIs , OS Version and carrier info)
- Applied Security Policy Information: Enterprise VPN Certificate Data for the user's employer

Example of exposed data on Pulse Secure Elasticsearch server

This personal information could be used for spear phishing or brute force authentication attacks on these customer organizations. Pulse Secure itself could also be a victim if an attacker chose to modify, remove or even ransom the Elasticsearch server data.

Some Appthority customers are also Pulse Secure customers and we have notified them of the HospitalGown threat and mitigated the risk of data exposure via the Appthority Mobile Threat Protection solution which includes policies that check for apps connected to insecure backend servers.

We have also been in contact with Pulse Secure regarding the leak. Pulse Secure has been very responsive in addressing and fixing the issue. We are revealing this information only for educational purposes and to avoid future incidents. The Pulse Secure app is no longer vulnerable to this type of data leakage at the time of the release of this report.

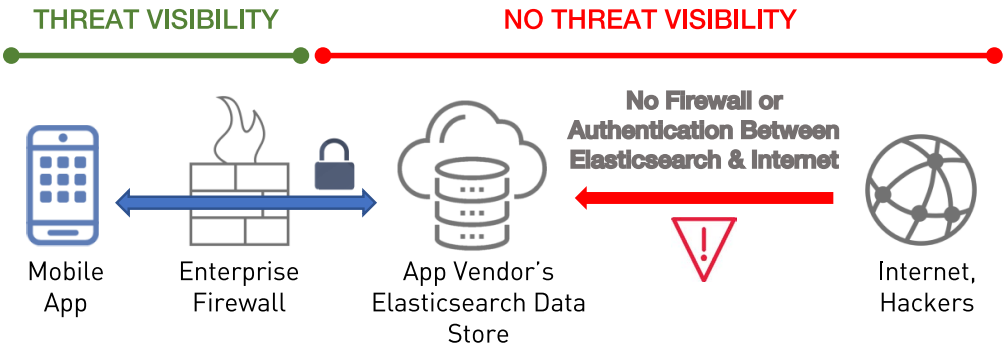
Example 2: Jacto Apps

Jacto is a global manufacturer of agricultural machinery such as self-propelled sprayers, tractor-mounted sprayers, and telemetry and autopilot systems for large-scale farms. Their machines connect to cellular and Wi-Fi networks to send information to dedicated servers via Jacto apps.

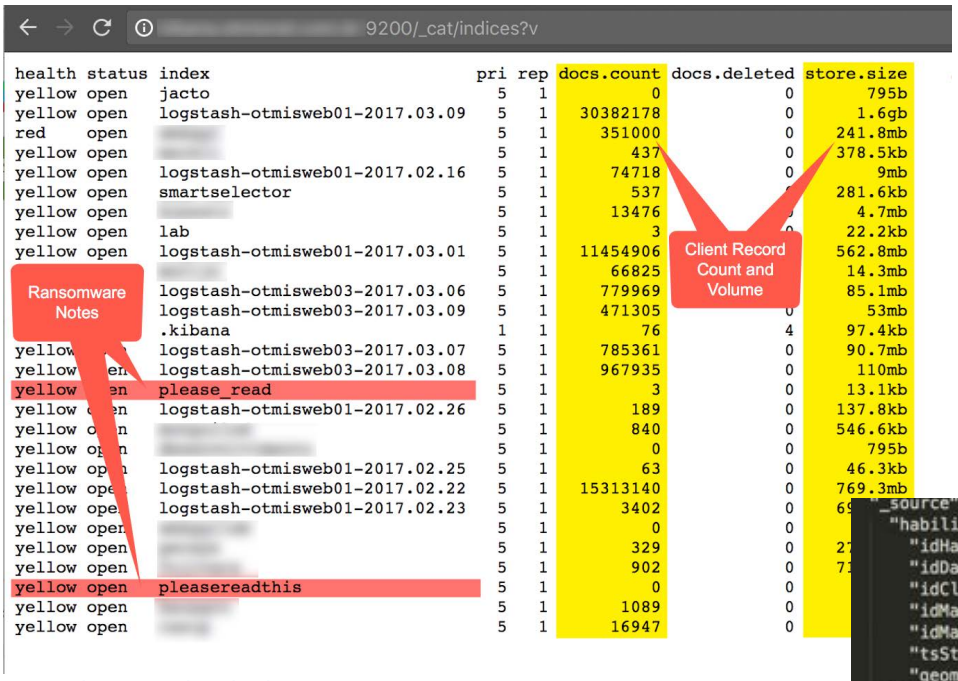
Appthority found three Jacto apps affected by HospitalGown: Jacto Smart Selector, OtmisNET and OtmisNET - Homologation. These apps help set autopilot parameters and monitor operational data about equipment in real-time through their on-board mobile devices to manage the operational efficiency of spraying. The data is not secured on its way to the data store and the backend servers for these apps are accessible to anyone with Internet access (see Figure 2 below), thus providing an avenue for leakage of all the tracked data.

Figure 2

Unsecured Mobile Backend No API: Jacto Case Study



Here we found 3.9 GB of exposed data and sampled 247 MB, (6.2% of the total Jacto customer data exposed).



Jacto Elasticsearch index list

Within the sample data set we discovered:

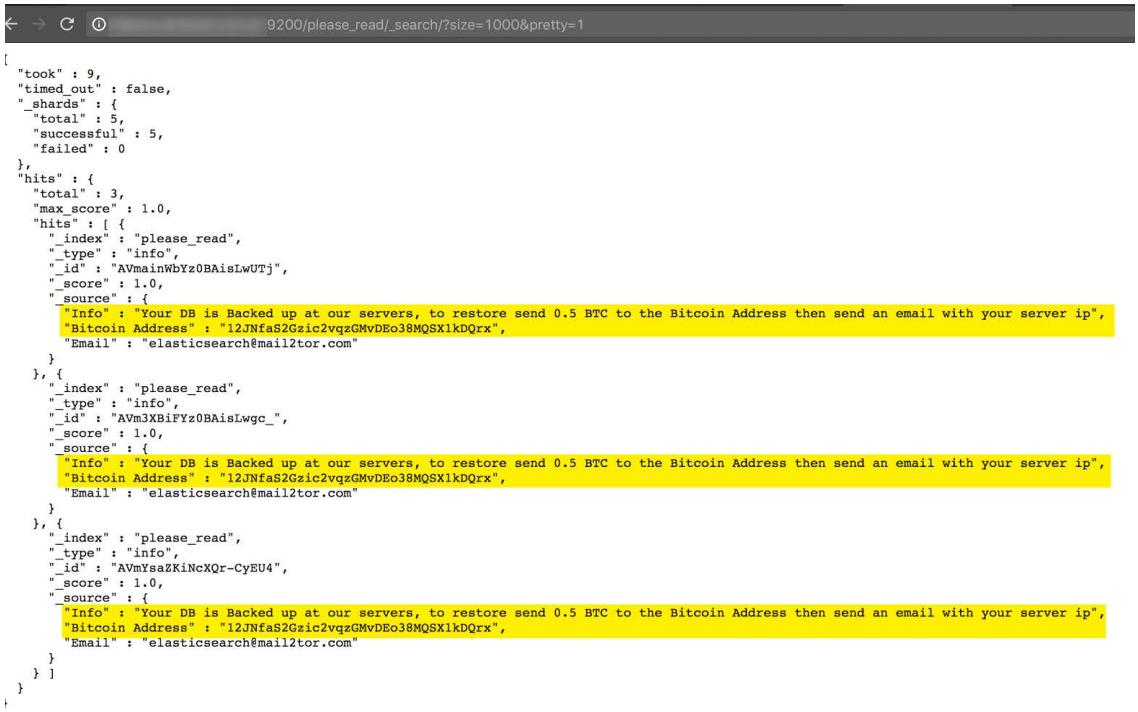
- Customer, partner and government agency records from over 10 countries
- PII data including 596 emails and 655 phone numbers
- Tractor/sprayer telemetry and operations data

In addition to the PII leakage already described, we were also able to track the tractors in real-time and access a range of information available on the server, including:

- Planned volume vs sprayed volume on the map
- The stop events of tractors, including location, time and reason for stopping, such as for meal or maintenance
- The speed of work performed and spray uniformity
- Humidity of the air at any point in the field where the spraying was performed
- Ambient temperature and engine temperature at the time of spraying
- Fuel consumption of the tractors

Customer's tractor telemetry data exposed on Jacto Elasticsearch servers

We recognize that, at first glance, tracking sprayer data may not look like a significant enterprise risk, but hackers can use this information in the same way Jacto's customers hope to – to perform analysis. In detail or in aggregate, this data could be used by a trader, competitor, nation-state, or bad-actor to estimate the cost and performance indicators of agricultural companies all over the world. From the recorded weather conditions and timing of spraying, agricultural experts might be able to predict the conditions of crops, and, consequently, calculate crop prices for commodities on the world's stock markets. A radical environmentalist may also be able to assess potential environmental damage caused by each agricultural firm, and create a valuable target list for sabotage or other nefarious activities. Since this data contains private customer data, market insights and competitive intelligence, it should be properly protected.



Ransom note detected in Jacto ElasticSearch (visible as an index)

Making this study in risk even more compelling, Appthority uncovered in our research that Jacto's exposed tractor data has already been held for ransom at least once by an attacker demanding bitcoins for payment. The attacker apparently took a copy of the entire data store, offering to delete their copy after being paid. Judging by the presence of the ransom index dating back to January 2017 on the server, Jacto may not have responded to the attacker's demands, or perhaps they paid, and failed to remove the reference to the attack from the data store.

Appthority has contacted Jacto developers regarding this important data leak but the company has not responded. Further, this data is still exposed as of this report's publication, although, evidence of the previous ransomware attack has been removed. Fortunately, none of our enterprise customers have been using any of the affected Jacto apps.

Conclusion

SUMMARY

In summary, the HospitalGown threat differs in three significant ways from most current mobile threats:

1. The threat does not require access to a company enabled device or user credentials – the threat is in the mobile application's backend infrastructure
2. The data exposure is at the administrative root level and not limited by user privileges – entire data stores are exposed to theft or ransomware.
3. The vulnerability is virtually impossible for security teams to detect without tools that can dig deep into applications and dynamically analyze the exposed backend infrastructure.

Every new mobile app that uses a back-end platform for data storage or analysis is a potential source of risk. Enterprises relying on software developers to properly code and configure the backend connections are exposed.

WHAT TO DO NOW

Because the risk is within the app provider's environment, security and mobility teams tasked with providing secure mobility for their companies may find they have very few direct options for protecting against HospitalGown data exposure.

However, Appthority recommends the following to improve your enterprise's security posture related to the HospitalGown vulnerability:

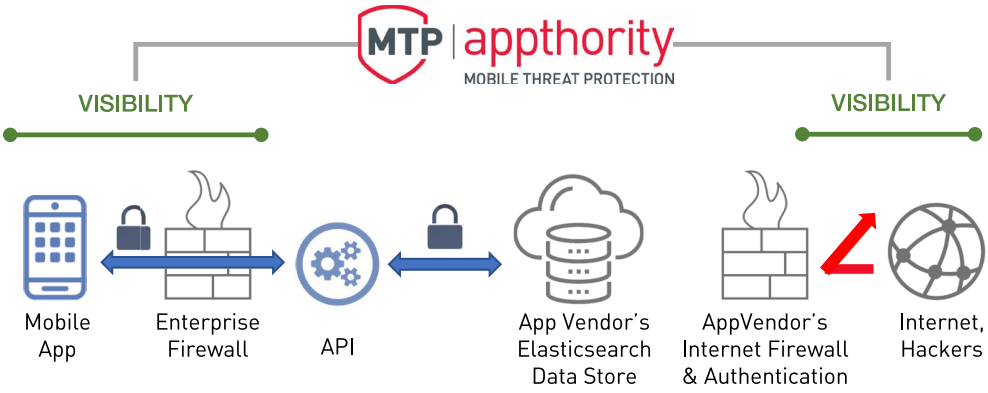
Know the apps in your environment

- Maintain a list of active apps and vendors so you know who has access to your data, particularly the sensitive data. Remove apps from your environment that may continue to store or access sensitive data when no longer in use. This applies not only to BYOD apps that employees download onto their devices, but to enterprise sanctioned apps that are available via an enterprise app store or Enterprise Mobility Management (EMM) platform.
- Review your current mobile app inventory immediately. If anyone in your enterprise environment is using the Pulse Workspace app, update to the latest app version immediately. If anyone in your enterprise environment is using the Jacto Smart Selector, OtmisNET, or OtmisNET - Homologation apps or any of the apps listed in Appendix A, contact the app developer immediately to request secure encryption of your data.

Know where your data is stored and get it secured

- Contact any vendors for which your enterprise uses a mobile app. Ask them for a letter from their legal department that confirms their app's backend is secured following best practices for the backend platform. Request that they limit the amount of personal information or site credentials stored on backend databases to the minimum needed for app functionality, and protect the data via encryption. (Note: This step is required by all organizations subject to the E.U. General Data Protection Regulation).

Properly Configured Mobile Backend with Appthority MTP Visibility



Deploy a Mobile Threat Defense Solution

To know the apps in your enterprise and the risks they pose and to ensure continuous detection and remediation of mobile threats such as HospitalGown, we strongly recommend all enterprises use a Mobile Threat Defense solution. At present, Appthority Mobile Threat Protection is the only such solution that can detect and remediate apps affected with the HospitalGown vulnerability. While EMM solutions provide an important foundational layer, effective enterprise mobile security requires a layered approach which is only achieved with EMM + Mobile Threat Defense.

Recommendations for Appthority Customers

Appthority initiated this research to ensure our customers' data was secure from backend data leakage. All customer apps have been analyzed and customers notified if they had any vulnerable apps in their environments.

Appthority recommends the following for maintaining data protection:

1. The Appthority Mobile Threat Protection (MTP) solution includes protection from HospitalGown threats. Simply apply the new threat indicator/behavior to your active policies to proactively detect new apps that are vulnerable to HospitalGown.

2. Utilize your Appthority policies to ensure the latest version of apps and operating systems are deployed.

3. If your enterprise is using a HospitalGown impacted app that you cannot blacklist or remove, contact the app developer or vendor to secure the backend server.

Note for all enterprises: Removing the vulnerable apps from mobile devices where the app has been used and synced with the backend server is not a full remediation option. Your data may still be retained and exposed on the backend server.

About Appthority

ABOUT APPTHORITY

About the Appthority Mobile Threat Team

Appthority's Mobile Threat Team (MTT) monitors and investigates mobile risks that pose a direct threat to mobile enterprises. Their goal is to provide research that educates and informs enterprises looking to protect their people, data, devices, apps, and networks from mobile risks. The MTT is comprised of top mobile security researchers and threat analytics managers who use their experience and expertise to develop best-in-class research insights. The team prides itself on delivering unique, accurate and practical perspectives, as well as security solutions, that help our enterprise audience understand the most impactful threats and address mobile risks.

About Appthority

Appthority is a pioneer in enterprise mobile security and the leader in the Mobile Threat Defense category. The comprehensive Appthority Mobile Threat Protection (MTP) solution helps customers keep their data private and secure from mobile device, app and network threats. More Fortune 1000 companies trust Appthority to secure their enterprises from mobile threats because Appthority delivers best-in-class mobile threat protection and unparalleled enterprise visibility and control of mobile risks. With Appthority, security teams are informed, employees are productive and enterprise data is kept private and secure.

APPTHORITY, INC.

535 Mission St., 20th Floor | San Francisco, CA 94105

FOLLOW US

Blog: https://www.appthority.com/mobile-threat-center/blog

Twitter: @appthority

CONTACT US

contact@appthority.com | +1 844-277-7475 | www.appthority.com

Appendices

APPENDIX A

Sample of Vulnerable Apps with Exposed Elasticsearch Servers

App Title	Package Name	Platform(s)
Asian Poker - Big Two	com.daidigames.banting	Android
Brave Frontier	sg.gumi.bravefrontier	Android
Capsa Royale: Capsa Susun, Big2, Pulsa Gratis	com.entertainment88.ios.capsaroyale	iOS
Capsa Royale: Susun,Pulsa Free	com.entertainment88.android.capsaroyale	Android
Chain Chronicle	sg.gumi.chainchronicleglobal	Android
EF Parents	com.ef.parents	Android / iOS
EFTrailblazers	com.ef.e1.students	iOS
Epoxy	com.socialjitney.couponz / com.davidgasparine.couponz	Android / iOS
Jacto Smart Selector	com.persys.smartselector	Android / iOS
JioCloud	jio.cloud.drive / com.jio.cloud.jiodrive	Android / iOS
Liputan6 - Berita Indonesia	com.woi.liputan6.android / com.kmk.liputan6	Android / iOS
OtmisNET	br.com.producao.otmisnet	Android / iOS
OtmisNET - Homologation	br.com.homologacao.otmisnet	Android / iOS
PulseSecure Workspace	net.pulsesecure.pws	Android / iOS
Raiders Quest RPG	com.alkemis.raidersquest	iOS
Seekmi	com.seekmi.customer	Android / iOS
Seekmi Vendor	com.seekmi.vendor	Android / iOS
Tiket.com - Flight & Hotel	com.tiket.gits / com.mobile.ios.Tiket	Android / iOS
Traveler Buddy	com.travelerbuddy.app	Android / iOS
Truco Free	com.blyts.trucolite.activities	Android
War Pirates	net.gogame.games.warpirates	Android / iOS
YourStory - Startup News India	com.yslabs.yourstory / com.YourStory	Android / iOS
Zonacitas: amor y encuentros	com.zc.zonacitas	Android
通达OA精灵2015	com.td.ispirit2015.ispirit	Android / iOS

APPENDIX B

Data Handling

To be responsible and ethical in its handling of the exposed data, the MTT limited its examination to metadata only when possible, and viewed less than one percent of the exposed data we characterize here. Our goal was not to view all the data but simply to be able to characterize the extent and nature of the data exposure. As Elasticsearch servers will list the size of each index in terms of records and size of the data, we did not need to download all the available data in order to know the size and scope of what was exposed.

Disclosures & Notifications

As part of Appthority's disclosure policy and in keeping with fair practices and a desire to improve enterprise security with our research and our products, Appthority notified a variety of parties affected by the HospitalGown vulnerability, and worked with them to provide information that would assist in resolving the issue. These included:

- Appthority customers: We alerted Appthority customers with apps in their environment containing the HospitalGown vulnerability. We also alerted customers whose data was exposed in the unprotected back-ends.
- Amazon: About unsecured Elasticsearch servers we identified hosted with Amazon Web Services
- Apple App Store and Google Play Store: About the affected apps we identified that send data to unsecured Elasticsearch servers
- App Developers: Of the all the apps identified as having the HospitalGown vulnerability, including technical details of what data is being sent and to where