My Signage Portal CMS Server Security Statement

Revision date: 08/09/2020

Cloud Server:

The CMS is hosted on an **Amazon Web Services (AWS) server** https://aws.amazon.com, using an **SSL encryption** method and an **ISO 27001 Security Certification**. **Nexpose & OpenVAS Vulnerability scan** takes place monthly, as well as the operating system (OS) for the server being updated and patched monthly by **AWS**.

The cloud uses '**AWS: EBS Provisioned IOPS storage (SSD)**', which protects data at rest and backs up all data on the server. This is secure storage and the second largest data centre in Amazon's global infrastructure. The server runs **Red Hat Enterprise Linux 7**, which is always kept up-to-date with the most recent the most version of the OS. Red Hat Enterprise Linux 7 is used instead of Microsoft Windows because it has less vulnerability issues than Windows as there are more known viruses compared to Linux. The Linux OS is more reliable and secure when preventing malicious software. The server also features a **Cloud Watch system**, which monitors the status of the server in real-time and sends updates to the engineers every minute. This also has a scalable computing power, allowing us to expand the server to suit our requirements.

The network administrators for the content management system (CMS) are fully trained engineers, and have at least 2 years' experience in our operations. Our very own development team also help maintain the server and processes. There is a **multifactor authentication** in place, which is used for the remote access. There is a locked office IP that allows the engineer in the office to access the server. This is a specific port for the engineer to maintain and manage the server. This is connected using an SSH server using the terminal emulator PuTTY.

Security:

The cloud data has to go through Amazon's **Key Management Service (KMS)** https://aws.amazon.com/kms/ before any data has been transferred to the screen/player. The KMS is a managed server that enables us to easily create and control the keys used for cryptographic operations. The service provides a highly available key generation, storage, management and auditing solution for us to encrypt data within the CMS. Events that breach security are logged and analysed by the KMS in real-time. Amazon's KMS monitors all the system resources and protects the confidentiality and integrity of data. KMS uses FIPS 140-2 validated Hardware Security Modules (HSM).

Amazon's EC2 Intrusion Detection System (IDS) and Intrusion Prevention System (IPS) offers key features to help protect EC2 instances. Any breaches are logged and analysed by the IDS system. These monitor the inbound and outbound data, as well as the system resources. Any attempted

intrusions will also be logged by the IPS. There are also engineers on standby 24/7 to respond to any vulnerability alerts. All security events are logged and reported. Any suspicious attempts are also reported immediately by AWS. The engineers and developers also have IT server maintenance training and server training, which is provided quarterly.

There are multiple security hardening procedures in place, such as trimming and restricting remote services on the AWS Server and minimisation of services the server is running. The CMS is also the only thing that the server hosts so there are no other processes being run on the server. It also features an **encrypted security key**, and uses **secured packages** when sending and receiving information. Each screen/player that is connected to the CMS will have a **security identifier**, which is unique for each screen/player. There is a **watchdog system** in place on the screen/player itself that will react to unwanted data or activity and begin restoring configurations from the server once abnormalities are detected. Finally, **two separate firewalls** are in place in front of all data going in and out of the server.

Ports and Protocols:

The server connects to the screen/player using the **HTTP+HTTPS web protocols**. HTTP is standard, but HTTPS is there for users that require it. If the HTTPS option is used, this **will encrypt the data both in and out with real-time generated security keys**, preventing any information being intercepted by others. On this same protocol, the server will transfer files via a **packet system**, and require the **screen/player to return a message** to the server every time it receives a part of the packet, allowing the next part to be sent.

Port **80 (or 8001)** using the HTTP protocol is used to **download media** from the server as well as configuration files. Port **4700** on the **UDP protocol** is used to send **remote commands** to the screen/player from the server. Port **4701** on the **TCP protocol** is used to let the screen/player **log into the server**, and also **register heartbeats** for monitoring purposes. Finally, ports **16732 and 16733** on the **UDP protocol** are used for local synchronisation of the content of multiple screens/players.

MY SIGNAGE PORTAL