SALT LAKE COUNTY
COUNTYWIDE INFORMATION TECHNOLOGY STANDARD
ON
PUBLIC FILE UPLOADS

Purpose –

It is the purpose of this information technology (IT) standard to provide a way to allow Salt Lake County to accept file uploads from the public while keeping County IT resources and systems safe from attacks (See examples in Appendix B)

Employees of Salt Lake County are expected to follow the *Public File Uploads Standard* established by the Information Technology Division. The Information Technology Division will monitor and enforce compliance with this standard.

Reference –

The standards set forth herein are provided in accordance with Countywide Policy 1400, which directs the Salt Lake County Information Technology Division to provide information technology standards. Also referencing the following:

All Countywide Information Technology Security Policies in the 1400 series
All Countywide Human Resource Policies

1.0 Scope

All Salt Lake County employees and anyone who uses County information technology resources or systems shall adhere to this Countywide information technology standard.

2.0 Terms and Definitions

Information Technology Resource(s) and/or System(s)
Computers, hardware, software, data, storage media, electronic communications (including, but not limited to, e-mail, fax, phones, phone systems, and voice mail), networks, operational procedures and processes used in the collection, processing, storage, sharing or distribution of information within, or with any access beyond ordinary public access to, the County's shared computing and network infrastructure.

MIME Type
A descriptor in the file metadata describes the file and its structure. For example, if the file is a JPEG image, the mime type should be "image/jpeg." If it's an Excel file, then the mime type will probably be "application/vnd.ms-excel".

3.0 Standard Guidance

Revision History
AUGUST 2021 – MLE

All Salt Lake County employees and anyone that uses a Salt Lake County IT resource or system will follow the County's Public File Uploads Standard as defined in Appendix A of this document.

4.0 Exceptions

Any exceptions to this standard must be explicitly approved in writing by the Salt Lake County Chief Information Officer or their designee.

5.0 Enforcement

Anyone found to have knowingly violated this standard may be subject to disciplinary action in accordance with County disciplinary policies.

Revision History
AUGUST 2021 – MLE

Appendix A
PUBLIC FILE UPLOADS

Files uploaded from the public directly to Salt Lake County property from either websites or applications should go through the following process:

1. Users must be authenticated before being allowed to upload files.
2. File types will be restricted per the needs of the project.
3. The maximum file size allowed will be restricted per the needs of the project.
4. Some files may require a secure file transfer system.
5. All file types and MIME types must be verified programmatically, and potential MIME Type Spoofing must be detected.
6. Scan and reject any files with Malware and scan for potential embedded threats.
7. Files will be stored on an external location from the web or application server.
8. The location of uploaded files will be secured appropriately based on the Security Team's standards.

File Upload Best Practices - https://blog.devolutions.net/2019/12/how-to-prevent-file-upload-vulnerabilities

Revision History
AUGUST 2021 – MLE

Appendix B
POTENTIAL RISKS

Cross Site Scripting (XSS) Attacks
A type of injection, in which malicious scripts are injected into otherwise benign and trusted websites. XSS attacks occur when an attacker uses a web application to send malicious code, generally in the form of a browser side script, to a different end user.

File Metadata Attacks
The path and file name can trick an application into copying the file to an unexpected location that could overwrite an important file and cause unexpected behavior. For example, an attacker could use control characters in the filename to trick the system into overwriting an important configuration file.

File Size Attacks
An unexpectedly large file can cause an application to overload or fail. For example, an attacker could use a botnet to trigger the simultaneous uploads of very large or very small files that result in legitimate requests not being fulfilled.

File Content Attacks
The content of the file is used to manipulate the behavior of the application. The outcome of this attack depends entirely on how the file is used and processed. For example, uploaded and executed Malware could be used to reveal a key that gives an attacker access to the system.

File Access Attacks
The access rules around uploaded files can be misconfigured, resulting in unauthorized access. For example, a misconfigured configuration could result in private files being accessible to the public.

Malware
Software that is specifically designed to disrupt, damage, or gain unauthorized access to a computer system.

Embedded Threats
Malicious code embedded within a file