Sertainty

UXP Identity

The UXP Identity plays a pivotal role in UXP Technology in how it functions to protect data.

The UXP Identity is necessary for generating a UXP Object. In a protected format, it houses the owner-specified access and mitigation policies and a list of permitted users for a dataset. These policies and the user list (*together, referred to as artifacts*) are required to create the unique protection scheme within the Object. Before an Object can be generated, a UXP ID must be present at the time and place when a dataset is protected.

After the Object is created, these provided artifacts are now embedded and protected in the Object and travel with it. These artifacts come into play again at an access attempt. As a self-managing and self-protecting entity, the Object's KCL Code (see UXP Object on the Sertainty Developer Portal) utilizes these embedded artifacts to assess as a comparison to its current environment when access is attempted. Trust is established and actively determined by the Object when the known embedded artifacts match EXACTLY to the current environment the Object resides in before it initiates authentication.

Outward Appearance

A UXP Identity is a special purpose UXP Object that is protected using the same protection scheme as a UXP Object. Unlike the Object, an Identity can't be accessed and contains no user-data. Thus, an Identity can never reveal its content, the access parameters for the dataset. It appears as an inert, binary file showing a *.iic extension and is unidentifiable unless proximal to UXP Technology Libraries. It simply looks and acts ordinary and nondescript on any O/S and can be easily designated as junk.

When in proximity to UXP Technology and an access attempt occurs, a single information page containing general public information or attributes (i.e. 30+ digit Identity number, ID name, date created, etc.) appears. No access parameters are seen or referred to in any way. These attributes along with additional unseen UXP Metadata provide verification details used by the Technology at the time when the UXP ID is used to protect a dataset. Without proximity, an access attempt reveals unreadable nonsense without disclosing the nature of the content or the context of the file extension.

Origin

Constructing a UXP Identity requires having the UXP Technology Libraries locally. Access parameters and a user list, defined by the data owner, are based on the data's security needs.

After UXP Technology Libraries are implemented, a UXP ID is generated either manually or automatically depending on the developer environment. The Identity is saved as a protected file with a *.iic extension on the OS.

The *.iic can be used in two forms: *.iic file on-disk or *iic structure in-memory.

Initial Function: The Front End of the UXP Object Generation

As stated above, the UXP ID must be present at the time and location when a dataset is protected. It provides essential artifacts that are incorporated into the process for constructing the protection scheme for the UXP Object.

During the Object generation, the process breaks the content (inclusive of the dataset and necessary UXP Metadata) into variable sized pages. Each file is uniquely protected within the UXP Object, where each file has its own protection scheme. The UXP Object protection scheme is formed using industry standard encryption (without modification) as well as a random set of proprietary algorithms that aren't predictable.

The encryption keys are created during the Object's creation process utilizing a random number generator that never exposes them and subsequently doesn't require sharing or reusing them.

Without the UXP ID artifacts, the protection scheme can't be constructed, and ultimately the Object can't be generated.

UXP ID Artifacts

The UXP ID artifacts are the owner-defined access and mitigation policies and the user list for the dataset. These artifacts fall into two categories: an access ruleset and User Definition(s), human, machine or process. A ruleset is a list of protocols that may define who, what, where, when and/or how a dataset is accessed. It can be one single rule or many depending on the owner's preferences based on the dataset's security requirements.

UXP Object Generation: UXP ID Artifacts as Building Blocks

When a UXP Object is generated, the UXP ID as a whole entity is NOT being used to construct the protection scheme. The UXP Technology Libraries at the initial stages of the process verify this entity as a UXP ID and extracts the artifacts (ruleset and User Definition attributes) and flushes the remaining elements. The artifacts are then incorporated in the protection scheme construction.

After the Object is created, the UXP ID artifacts are embedded and uniquely protected within the Object and now travel with it unseen where ever it goes.

Authentication

When the UXP Object is residing at its expected destination and is "activated", the KCL Code (see UXP Object on the Sertainty Developer Portal)) is covertly assessing the environment to determine trust before permitting authentication. The KCL Code compares the current setting and its specifics to the known access ruleset and unique machine profile details (artifacts provided by the UXP ID at the Object's generation) that travel embedded within the Object. The embedded artifacts MUST match without ANY deviation to the current environment before authentication is initiated. The KCL Code can actively deny access if it deems the environment untrustworthy.

When trust is established and environmental details coincide identically with the UXP ID artifacts embedded in the Object, the KCL Code permits authentication and the user dataset may be accessed.

Email us today:
tech-support@sertainty.com