Security Considerations for Administrative Facets

Exposing administrative functionality naturally makes a program vulnerable, therefore it is important that proper precautions are taken. With respect to the default functionality, the `Properties` facet could expose sensitive configuration information, and the `Process` facet supports a `shutdown` operation that opens the door for a denial-of-service attack. Developers should carefully consider the security implications of any additional administrative facets that an application installs.

There are several approaches you can take to mitigate the possibility of abuse:

- Disable the administrative facility

  The administrative facility is disabled by default, and remains disabled as long as its prerequisites are not met. Note that IceGrid enables the facility in servers that it activates for the following reasons: You could disable a facet using filtering, but doing so may disrupt IceGrid's normal operation.

  Unknown macro: 'ul'

- Select a proper endpoint

  A reasonably secure value for the `Ice.Admin.Endpoints` property is one that uses the local host interface (`-h 127.0.0.1`), which restricts access to clients that run on the same host. Incidentally, this is the default value that IceGrid defines for its servers, although you can override that if you like. Note that using a local host endpoint does not preclude remote administration for IceGrid servers because IceGrid transparently routes requests on `admin` objects to the appropriate server via its node. If your application must support administration from non-local hosts, we recommend the use of SSL and certificate-based access control.

- Filter the facets

  After choosing a suitable endpoint, you can minimize risks by filtering the facets to enable only the functionality that is required. For example, if you are not using IceGrid's server activation feature and do not require the ability to remotely terminate a program, you should disable the `Process` facet using the filtering mechanism.

- Consider the object's identity

  The default identity of the `admin` object has a UUID for its category, which makes it difficult for a hostile client to guess. Depending on your requirements, the use of a UUID may be an advantage or a disadvantage. For example, in a trusted environment, the use of a UUID may create additional work, such as the need to add an interface that an administrative client can use to obtain the identity or proxy of a remote a `dmin` object. An obscure identity might be more of a hindrance in this situation, and therefore specifying a static category via the `Ice.Admin.InstanceName` property is a reasonable alternative. In general, however, we recommend using the default behavior.

See Also

- The admin Object
- The Properties Facet
- The Process Facet
- The Administrative Object Adapter
- IceGrid and the Administrative Facility
- Filtering Administrative Facets
- Ice Administrative Properties
- IceGrid
- IceSSL