User Identification

Some applications can operate in the presence of anonymous authentication, which essentially allows applications to know that authentication occurred, often with some additional non-identifying information about the user, but not to recognize the user's unique identity. Usually this is a qualified "not", since IP addresses, browser fingerprinting, or other information can be used to correlate, but this usually only works for "imperfect" correlation. Good enough for ads and data mining, but usually not enough for real security.

Most applications need some form of persistent user identification (persistent meaning not limited to one session with the application, and usually lasting for an extended number of sessions). There are a number of useful properties one can use to discuss and contrast different identifiers:

- Persistence
- Change Frequency
- Auditability/Traceability
- Reassignment
- Opacity/Directionality/Correlatability
- Readability/Displayability
- Portability

Aside from a taxonomy:

- Use within Applications
    - Direct use/storage
    - Locally keyed users linked to one or more external identifiers
    - Display, user enumeration/selection
    - Resource sharing/invitation of users

- Architectural Considerations
    - Mapping of identifiers across proxies/gateways
    - Dealing with multiple identifiers

- E-Mail Addresses: Friend or Foe