Autenticació vs Autorització

S'anomena "autenticació" al procés de verificar una identitat. Aquest procés implica la identificació d'un element (que sol ser un usuari però també pot ser un programa o, en el cas d'interaccions per xarxa, fins i tot una màquina) per part del sistema verificador (que pot ser un programa local com *gdm*, *login*, *sudo* o *su* o un servidor remot com SSH, POP/SMTP, etc)

Hi ha moltes maneres d'autenticar: mitjançant contrasenyes simples, contrasenyes d'una sola vegada (OTP), certificats,, exploracions biomètriques, etc ... totes aquestes maneres s'anomenen "credencials" en general. El tipus exacte de credencial requerida estarà definit pel mecanisme d'autenticació que el sistema verificador tingui configurat. Els mecanismes més habituals (i que es poden combinar entre sí) són:

*__Basat en contrasenyes simples__: Permet autenticar usuaris si aquests proporcionen un nom i contrasenya reconeguts com a vàlids pel sistema verificador. Es basa en quelcom que l'usuari *sap*

*__Basat en contrasenyes OTP__: A one-time password (OTP) is an automatically generated numeric or alphanumeric string of characters that authenticates the user for a single transaction or login session. An OTP is more secure than a static password because it can't be reused. Very often OTPs are involved in some kind of 2FA ("Two Factor Authentication") or MFA ("Multiple Factor Authentication") process where, after user providing a simple password, he/she must generate the OTP in his/her pre-tied hardware device (which can be an specialized one or simply a mobile phone using some specific app - although if using the SMS's one then the OTP is not generated on device but received from third-party server-) to then provide it too so that finalize the authentication. D'aquesta manera, les autenticacions 2FA es basen en quelcom que l'usuari *sap* (la contrasenya simple inicial) més quelcom que l'usuari t*é físicament* (el dispositiu hardware/app de mòbil on es rep la OTP). Per més informació podeu consultarl'article: https://www.eff.org/deeplinks/2017/09/guide-common-types-two-factor-authentication-web i també https://sec.eff.org/topics/two-factor-authentication

> NOTA: Una "segona generació d'autenticació 2FA és la que fa ús de claus UF2 ("Universal 2 Factor") per tal d'evitar l'ús d'apps mòbils a l'hora de gestionar les contrasenyes OTP (degut a les seves possibles vulnerabilitats). Les claus UF2 són dispositius hardware de tipus USB (o també inalàmbrics de tipus NFC) que fan ús de criptografia asimètrica i es poden configurar per autenticar-se automàticament contra, sobre tot, aplicacions web (correu, xarxes socials, etc). More specifically, when a user tries to access an (previously configured accordingly) account, after first entering static password he/she should glance at the device and enter the displayed 2FA code back into the site or app. Other versions of keys can automatically transfer the 2FA code to the recognized site's domain name when plugged into a computer's USB port (if this domain name has been introduced previously into key) without nearly any manual intervention. Aquestes claus poden estar desenvolupades per diferents fabricants (un dels més coneguts és Yubico, que fabrica els seus "yubikeys") però que tots segueixen el mateix estàndar UF2, estandaritzat per la FIDO Alliance (un conglomerat d'empreses entre les quals es troben Google, Facebook, Microsoft, etc).

> NOTA: Un altre exemple d'autenticació 2FA (aquest cop sense que intervinguin OTPs, però) és la proporcionada pels caixers automàtics perquè es basa en quelcom que l'usuari *té físicament* (la tarja bancària) i quelcom que l'usuari *sap* (el pin).

*__Basat en certificats (de client):__ Permet autenticar la màquina de l'usuari. Funciona així: aquesta màquina envia un certificat (que és un fitxer criptogràficament vinculat a dita màquina) a través de la xarxa a un servidor que haurà de confirmar la validesa del certificat i, per tant, la identitat de la màquina en qüestió. Es basa en quelcom que la màquina de l'usuari *guarda*

> NOTA: El protocol TLS, molt utilitzat a Internet per assegurar les comunicacions dels navegadors amb servidors web (entre d'altres), utilitza un mecanisme molt similar també basat en certificats però en aquest cas de servidor.

__*Basat en "smart cards":__ Variant "hardware" del mecanisme anterior. La "smart card" (o, més habitual actualment, una "app" de mòbil) emmagatzema el certificat; quan l'usuari insereix un determinat "token" en el sistema, aquest llegirà el certificat i concedirà l'accés. Es basa en quelcom que l'usuari *té físicament*

*__Biomètric__: Existeixen diferents mètodes, com l'empremta dactilar, la forma de la mà, el patró de l'iris, la veu, etc. A https://cromwell-intl.com/cybersecurity/authentication.html#authtool_biometric es pot consultar una llista de fabricants de components biomètrics. Es basa en quelcom que l'usuari *és*.

*__Mitjançant Kerberos:__ Sistema de credencials de curta durada i multi-servidor. Funciona així: l'usuari primer presenta al servidor Kerberos unes credencials pròpies (normalment, són un nom d'usuari i contrasenya, però poden ser de qualsevol dels altres tipus esmentats: certificats, biomètrics ... o una combinació d'aquests), per identificar-se i així poder rebre d'aquest servidor la credencial de curta durada, anomenada TGT. Aquest TGT s'utilitzarà a partir de llavors per accedir automàticament a altres serveis, (com ara servidors de correu electrònic, de carpetes compartides, etc). La "gràcia" de l'autenticació fent servir Kerberos és que permet realitzar només un procés d'autenticació (al principi, per demanar el TGT) perquè a partir de llavors es reaprofita aquest procés per accedir a múltiples servidors diferents de forma totalment automatitzada (mentre duri el període de validesa del TGT, és clar.). Aquest fet és el que s'anomena "Single Sign-On" o SSO i s'utilitza principalment en entorns corporatius.

NOTA: Cal tenir en compte que el SSO posa "tots els ous en un cistell": si només una contrasenya és robada o endevinada, llavors *tots* els recursos i identitats estan compromeses. Cal tenir un equilibri, doncs, entre conveniència i seguretat.

Tal com hem dit, el sistema verificador ha de tenir configurat un determinat mecanisme d'autenticació. La tasca d'aquest mecanisme, sigui quin sigui aquest, però, sempre és la mateixa: comparar cada credencial presentada amb les que tingui emmagatzemades prèviament en una (o diverses) "bases de dades": si ambdues credencials coincideixen, la comprovació serà reeixida i la part sol·licitant s'autenticarà.

En el cas concret de fer servir l'autenticació basat en contrasenyes, les "bases de dades" que consulta el sistema verificador (que és on, en definitiva, estan guardats els noms i contrasenyes vàlids) es poden trobar:

*__Localment dins el propi sistema verificador__. Normalment en forma de fitxers de text (com és el cas del fitxer "/etc/shadow" a Linux) o de registre binari (com és el cas de Windows)

*__En un servidor remot que sigui accedit quan calgui pel sistema verificador.__ Aquest servidor remot pot ser de diferents tipus segons els requisits que necessiti el nostre sistema d'autenticació particular però bàsicament han de proporcionar un sistema d'emmagatzemar noms i contrasenyes. Els més comuns són:

-Un servidor LDAP
-Un sistema gestor de base de dades relacional (SGBD)
-Un servidor Kerberos (que proporcionarà TGTs)
-Un servidor RADIUS

NOTA: Tant els servidors LDAP con els SGBDs poden contenir, a més de la informació estrictament relacionada amb l'autenticació (noms i contrasenyes, bàsicament), diferent informació extra relacionada amb els usuaris autenticats (com ara dades personals: telèfon, email, etc o dades de sistema: ruta de la carpeta personal, shell preferit, etc, etc). De fet, fins i tot es podria combinar l'ús de Kerberos (o RADIUS) amb un altre servidor LDAP o SGBD, delegant així el procés d'autenticació només en Kerberos (o RADIUS) i fent servir la base de dades LDAP/relacional només de font d'informació complementària. RADIUS, per la seva banda, també incorpora certes funcionalitats relacionades amb processos d'autoritzacions, principalment relacionades amb l'accés a xarxes

---

S'anomena "autorització", d'altra banda, al procés que determina què pot fer i/o on pot accedir l'element autenticat. Un cop un determinat element ha sigut identificat (per exemple, un usuari), el sistema determinarà, amb les dades que n'obtingui d'aquest usuari (provinents de diferents fonts), a quins recursos (és a dir, a quins programes, carpetes i fitxers, hardware, etc) estarà autoritzat a entrar / connectar / llegir / modificar / escriure / eliminar / executar, etc i a quins no. Així doncs, cal distingir entre autenticació i l'autorització perquè són dos processos separats.

Les fonts d'informació que pot fer servir el sistema per construïr l'esquema d'autoritzacions d'un determinat usuari autenticat són moltes i molt variades. Des del sistema de permisos de fitxers i carpetes clàssic passant per les ACLs i les "capabilities" fins arribar a esquemes més complexos com els MACs SELinux (o AppArmor) o com els subsistemes proporcionats per *sudo* o pel framework Polkit, entre molts d'altres. Tot això combinat amb el possible ús de servidors remots LDAP (o relacionals o RADIUS) per completar tota aquesta informació.