Autentizace a autorizace (způsoby autentizace, cookie, session proměnné)

Autentizace

Autentizace slouží k jednoznačnému určení uživatele vstupujícího do systému. Ve své podstatě jde o ověření totožnosti uživatele – kvůli bezpečnosti i přizpůsobení se takovému uživateli.

Většinou je proces autentizace zahájen snahou o přistoupení k něčemu, k čemu je přístup omezen – následovaný výzvou k zadání dvojice uživatel-heslo. Po zadání hodnot může proběhnout samotná autentizace (obvykle porovnání hodnot s hodnotami v databázi). Pokud vstupní data souhlasí, požadovaná data jsou uživateli zobrazena.

Autorizace

Autorizace je proces ověření přístupových oprávnění uživatele vstupujícího do systému nebo uživatele snažícího se využít nějakou specifickou funkci systému. Může navazovat na autentizaci. Podstatou autorizace je ověřit, zda daný uživatel má oprávnění požadovanou akci provést.

Oprávnění se mohou vázat na uživatele nebo na skupinu, jíž je uživatel členem – záleží na aplikaci. Obvykle jsou data součástí informací o uživateli uložených v databázi.

Způsoby autentizace

Protokol HTTP

Samotný protokol HTTP umožňuje autentizaci uživatele. Tento způsob se obvykle nastavuje přímo na serveru, k němuž je přistupováno. Při snaze na takový server vstoupit je prohlížeči odeslán kód 401 Unauthorized, prohlížeč chybu rozpozná a automaticky zobrazí výzvu k autentizaci uživatele prostřednictvím formuláře.

Informace týkající se autentizace jsou přenášeny v hlavičce WWW-Authenticate. Ta obsahuje mimo jména a hesla uživatele také například metodu kódování uživatelem zadávaných informací (Basic/Digest). Digest je bezpečnější, ale není podporován všemi servery ani prohlížeči – vytvoří se náhodný řetězec (klíč) a ten se odešle v hlavičce také. V hlavičce může také být uložena oblast, pro kterou autentizace platí. Různé části serveru (složky) mohou být jinými oblastmi.

Databáze

Nejběžnější způsob autentizace. V databázi jsou o uživateli uložené údaje a při přihlašování jsou z databáze údaje o daném uživateli načteny. Pokud se hash hesla shoduje s hashem uloženým v databázi, je autentizace úspěšná.

.htaccess a .htpasswd

Je-li na serveru povoleno použití souboru .htaccess, je možné jej využít také k autentizaci uživatele. Nastavení souboru .htaccess se vztahují vždy na složku a podsložky složky, ve které je soubor je.

V případě autentizace prostřednictvím těchto dvou souborů je nutné tyto dva soubory vytvořit a naplnit potřebnými daty. V souboru .htaccess je uložena cesta k souboru .htpasswd, název prostoru (jako u HTTP autentizace) a typ ověření (Basic/Digest).

Soubor .htpasswd se doporučuje umístit do kořenové složky, protože tam bývá nejlépe chráněn (na různé podadresáře se už mohou vztahovat různá jiná nastavení souborem .htaccess a mohlo by být např. možné si .htpasswd vzdáleně zobrazit).

V .htpasswd by hesla měla být uložena formou hashe vytvořeného PHP funkcí `Crypt()` ve tvaru username:hash. Každý uživatel se nachází na novém řádku.

Na straně klienta funguje ověřování stejným způsobem jako autentizace protokolem HTTP.

Cookie

Cookies jsou malé textové informace uložené v počítači klienta. Soubory cookie jsou obvykle vytvářeny serverem, který je může získat zase zpět. Vytvářeny jsou z toho důvodu, že je protokol http je bezestavový. Uložíme-li si něco do serverové proměnné ($_GET, $_POST), do URL adresy či někam jinam, co je součástí http přenosu, data se po provedení požadavku ztratí.

Cookies jsou při každém požadavku na server, který cookie vytvořil, na takový server posílány. U každé cookie lze nastavit několik vlastností, mezi které patří mj. *jméno cookie* (jediný povinný parametr), *obsah cookie* (její hodnota) a *expirace cookie* (platnost, kdy bude z klienta smazaná, udává se unixový čas).

Pokud není čas expirace uveden, obvykle je cookie smazána při zavření prohlížeče.

V PHP se pro vytvoření cookie nachází funkce `setcookie()`.

Session proměnné

Session funguje podobně jako cookies s tím rozdílem, že jsou soubory uloženy na straně serveru. Session je z v PHP zahájena zavoláním funkce `session_start()`, čímž je na serveru připraven soubor pro danou session a do počítače klienta je uložena cookie s ID takové session (obvykle je nazvána SESSIONID). Sessions jsou uložené v serverové proměnné $_SESSION.

Obsah session bývá vytvořen při nějaké akci, typicky při přihlášení uživatele. V takovém případě se do session uloží například informace o uživateli z databáze, aby nebyla databáze zbytečně při každém požadavku zatěžována (údaje o přihlášeném uživateli nemusí být z databáze již načítány).

Může se pak objevit například $_SESSION["username"] či $_SESSION["permissions"], k nimž lze takto přistupovat (načítat je či upravovat).

Při odhlášení uživatele je nutné session řádně ukončit, aby se k datům již nebylo možné dostat. Nejvhodnějším řešením je odstranění všech session proměnných funkcí `unset()` a následně session ukončit funkcí `session_destroy()`.