Икономически университет – Варна

Факултет “Информатика”

РЕФЕРАТ

Дисциплина: **„ Езици за програмиране”**

Тема: **Алгоритмични проблеми при внедряване на двуфакторна автентикация в уеб приложения**

Докторант: **Петър Димитров**

Докторска програма: **„Информатика”**

Катедра: “Информатика”

Научен ръководител:

доц. д-р Павел Петров

ВАРНА

2018

Съдържание

Въведение

В днешно време термините „потребителско име”, „парола” и „токен” са общоприети и популярни, което е в резултат на изключително високата степен на дигитализация не само на публичните услуги, но и при услугите за съхраняване на личните данни. В исторически план терминът „парола” се среща в древни текстове - например Библия, Книга Съдии Израилеви или в приказките - например приказката "Али Баба и 40-те разбойници". В контекста на информационните технологии „пароли” започват да се използват след създаването на първите компютри, работещи в режим времеделение в началото на 60-те г. на миналия век, когато множество потребители достъпват един и същ ресурс, при което системата използва парола, за да ги идентифицира. Тезата, че комбинация от символи, помнена от крайния потребител, може да осигури сигурна идентификация на потребителя към услуга, е поставена под съмнение, след като през 1984 г. Кенет Вайс патентова идеята си „Метод и апарат за положително идентифициране на индивид”[1], която поставя началото на т.нар. "втори фактор" автентикация. При нея освен паролата, която трябва да се запомни, потребителят трябва да притежава някакъв вид устройство.

Целта на реферата е да разгледа алгоритмичните предизвикателства пред разработчиците при внедряване на двуфакторна автентикация в уеб приложения чрез средствата на програмния език PHP. За постигането на целта са поставени следните задачи:

1. Да се анализира необходимостта от повишаване на сигурността в уеб.
2. Да се разгледат методи и подходи, които са широко разпространени
3. Да се направи критичен анализ на използваните досега методи.
4. Да се внедри универсална двуфакторна автентикация в PHP-

[1] Kenneth P. Weiss, Method and apparatus for positively identifying an individual, USPTO, 1984, <https://patents.google.com/patent/US4720860> (достъпено 05.06.2018 г.)

базирано уеб приложение.

4. Да се анализират ограниченията и да се разгледат възможностите за бъдещо развитие.

Проблеми при съвременното прилагане на двуфакторна автентификация

В периода от 1984 г. до днес устройствата за двуфакторна автентикация стават все по-разнообразни като по наши наблюдения най-голям принос за тяхното развитие имат банките, поради предлаганите от тях електронни услуги. Причината за това е, че потребителите на активни онлайн банкирания трябва да са обезпечени с максимална степен на сигурност[1]. Там въпросите, свързани с управление на информационната безопасност са изведени на преден план. В днешно време са разработени методи, чрез които не е нужно специално токен устройство, а само и единствено смартфон с цел максимално да се улесни потребителя, без това да води до намаляване на сигурността. Два от най-популярните методи за подобен вид двуфакторна автентикация са: първо, автоматично генериран код, изпратен чрез SMS и второ, специално приложение за автентикация, генериращо кодове по определен алгоритъм на определен интервал от време. Вторият начин до голяма степен е аналог на хардуерния токен, който все още е доста разпространен начин за автентикация.

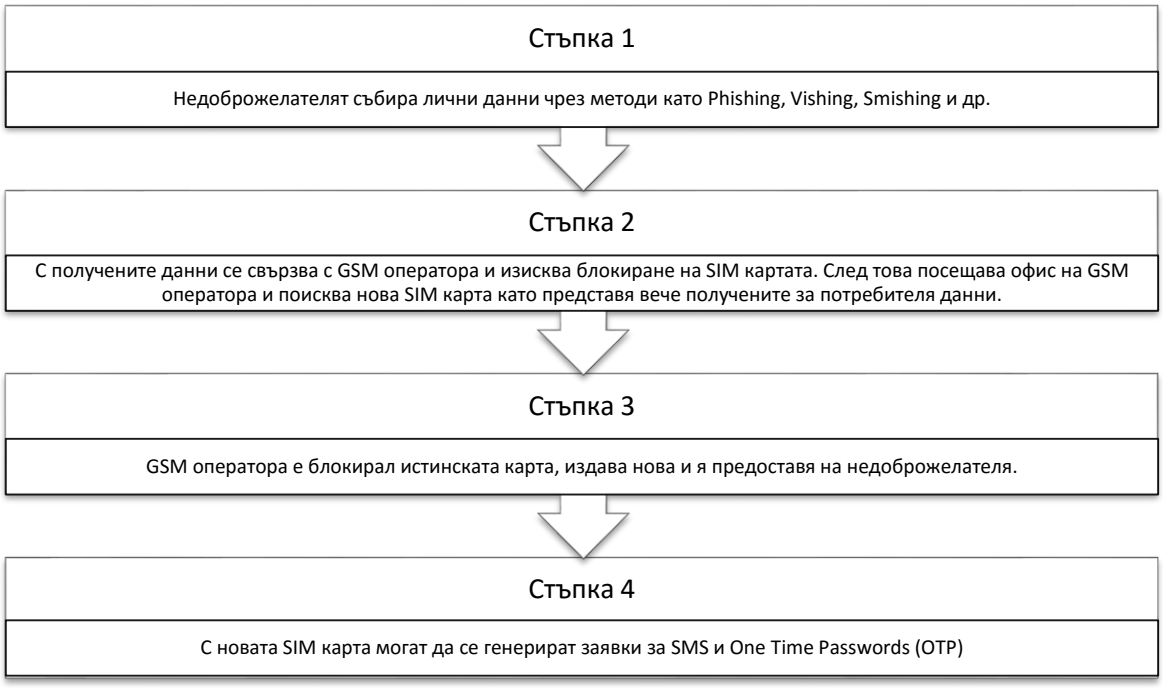
Генерираният от сървъра код обикновено се използва при първоначално идентифициране на потребителя, но има случаи, в които се изпраща чрез SMS нов код за всяка операция.Това води след себе си до редица проблеми, част от които остават скрити за крайния потребител, а именно изключително големия обем от алгоритмични изчисления и разбира се факта, че SMS услугата не е безплатна. Подобен начин за автентикация

[1] Дражев, С., Парушева, С., Петров, П. Стратегия за защита на уеб базирани банкови информационни системи. Сборник научни трудове: Посветен на 105-год. от рождението на пионерите на компютърната техника Джон Атанасов и Джон Фон Нойман, Шумен: Унив. изд. "Е. К. Преславски", 2, 2009, с.29-35.

крие и редица рискове, които карат експерти по сигурността да съветват потребителите да не използват SMS за втори фактор автентикация. През последните години зачестяват случаите на т.нар. Sim Swap. В държави като Великобритания[1] и САЩ[2] например е възможно недоброжелателен потребител да поиска от телекомуникационната компания смяна на мобилния телефон без знанието на жертвата, след като се сдобие с някакъв вид лични данни (например последните цифри на номера на социалната осигуровка). По този начин SMS съобщението би пристигнало директно на телефон, до който хакера има достъп (фиг.1).

Над 80% от потребителите в САЩ притежават смартфон[3] и в Европа делът им също е висок (виж фиг. 2). Можем да приемем приложенията за автентикация за естествено подобрение на кода, изпращан чрез SMS по няколко причини – количеството ресурс, отделен от доставчика на платформата значително намалява, и не на последно място се предлага удобно решение на потребителите, при което не е нужно те да получават SMS съобщение с код, който да въвеждат. Нещо повече – приложенията предлагат възможност за добавяне на повече от един автоматично генериращ се код в случай, че потребителят използва различни платформи, имплементиращи този начин на автентикация.

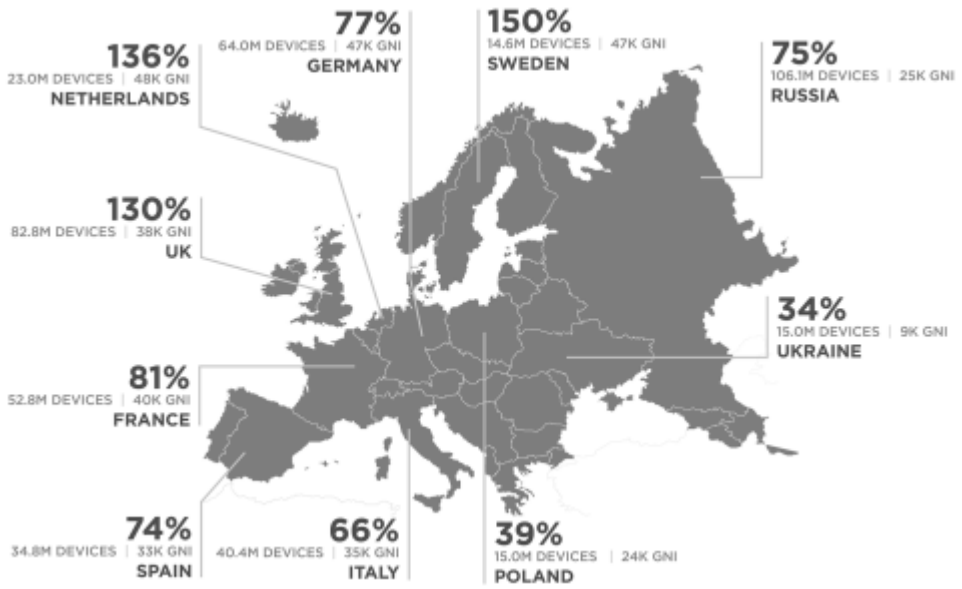
[1] Anna Tims, 'Sim swap' gives fraudsters access-all-areas via your mobile phone, The Guardian, 26.05.2015, <https://www.theguardian.com/money/2015/sep/26/sim-swap-fraud-mobile-phone-vodafone-customer> (достъпено 05.06.2018 г.)
[2] AT&T SIM-Card Switch Scam, Department of State's Division of Consumer Protection, 2014 <https://www.dos.ny.gov/consumerprotection/scams/att-sim.html> (достъпено 05.06.2018 г.)
[3] Lella, Adam U.S. Smartphone Penetration Surpassed 80 Percent in 2016 <https://www.comscore.com/Insights/Blog/US-Smartphone-Penetration-Surpassed-80-Percent-in-2016> (достъпено 05.06.2018 г.)



Фиг. 1. Схема на прилагане на "Sim Swap"

За съжаление методът, използван от тези приложения, макар доста интуивен за използване, крие някои опасности[1]. Ще разгледаме в детайли използването на двуфакторна автентикация в уеб приложение и идентификацията от страна на потребител с приложението на “Google Authenticator”. Важно е да се отбележи, че при повечето уеб приложения двуфакторната автентикация предоставя допълнително ниво на сигурност и не е задължителна.

[1] Stanislav, Mark Two-Factor Authentication, IT Governance Publishing, 2015



Фиг. 2. Използване на смартфони в Европа

Процесът по включване на двуфакторната автентикация се прави в няколко стъпки, като в първата потребителят се подканва да инсталира приложението “Google Authenticator” на смартфона си и чрез него да снима генерираното от платформата изображение (фиг.3).

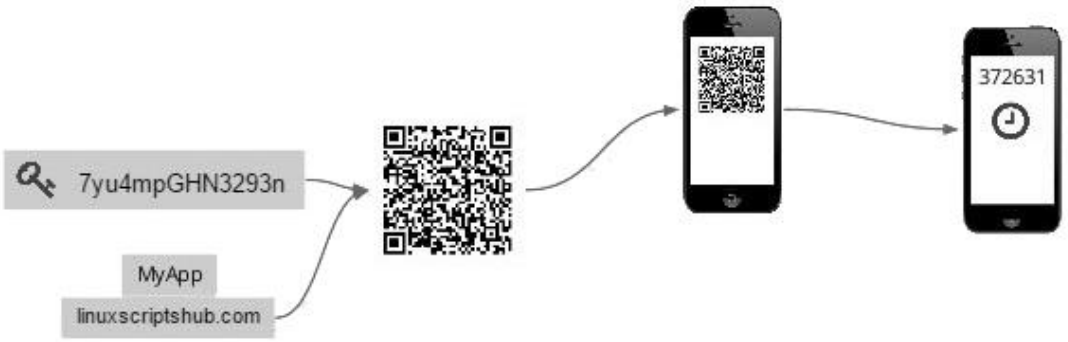
Фиг. 3. Пример за двуизмерен QR код, използван в приложението “Google Authenticator”

За генериране на това изображение могат да се използват различни методи, като за целите на тази разработка се спираме на PHP клас с отворен код[1]. Този клас се използва както за генерирането на QR код, така и за удостоверяването на потребителя след като е добавил двуфакторната автентикация към акаунта си.

[1] https://github.com/PHPGangsta/GoogleAuthenticator

```php
<?php
require_once 'include/GoogleAuthenticator.php';

// Инициране на PHP класа
$ga = new PHPGangsta_GoogleAuthenticator();

// Дефиниране на празен масив с данни
$result = array();

// Генерираме случаен низ и го записваме в масива $result
$result['2fa_secret'] = $ga->createSecret();

// Генерираме изображение, подавайки като аргументи ID на потребителя,
// случайния низ и заглавието на проекта ни
$result['qrCodeUrl'] = $ga->getQRCodeGoogleUrl($_SESSION['user'],
$result['2fa_secret'], 'WebStudentDB');

// За целите на идентификацията генерираме тестов код
$result['oneCode'] = $ga->getCode($result['2fa_secret']);

// проверяваме дали генерирания код е валиден спрямо генерирания низ
$checkResult = $ga->verifyCode($result['2fa_secret'], $result['oneCode'], 2);
```

Ако променливата `$checkResult` има стойност **`true`**, в базата данни съхраняваме стойността на `$result['2fa_secret']` за конкретния потребител. Важно е да се отбележи, че случайния низ представлява 16-символен BASE32 низ. При сканирането на това изображение в приложението се добавя нов елемент с автоматично генериращи се кодове и тук е момента в който потребителят трябва да уведоми сайта, че кодът е сканиран като въведе току-що генерирания от “Google Authenticator” код в уеб приложението. На заден план уеб приложението извършва редица изчисления на база 16-символния низ и въведения код и ако изчисленията върнат положителен резултат, се включва двуфакторната автентикация за конкретния потребител, запазвайки в базата си данни 16-символния низ за да се използва за последващо идентифициране (виж фиг. 4). В този момент потребителят е включил двуфакторна автентикация и не е наясно с това, че ако някой успее да достигне до този 16-символен низ може да генерира същия QR код, да го сканира в своето копие на “Google Authenticator” и да получи абсолютно същите кодове за двуфакторна автентикация по всяко време.



Фиг. 4. Принципна схема на работа с "Google Authenticator"

Универсална двуфакторна автентикация

През 2012 г. компанните Yubico и Google създават U2F – Universal 2 Factor на концептуално ниво с идеята този нов метод за автентикация да реши повечето от трудностите и проблемите, които присъстват във вече утвърдените методи за двуфакторна автентикация. Принципите, на база на които е създаден модела на U2F, съвпада с принципите, изложени от формирания по същото време FIDO Alliance – организация, възникнала с идеята да създава стандартизирани услуги за автентикация и тяхното популяризиране. Към края на 2014 г. протоколът U2F е публично оповестен и множество компании започват разработването на съвместими към стандарта, създаден от FIDO, устройства.

Съществуват значителни разлики между разгледаните методи за двуфакторна автентикация и U2F метода. Някои от тях са очевидни, но други остават скрити за потребителя, който използва U2F.

Една от основните разлики е премахването на нуждата потребителят да въвежда код, бил той генериран през Google Authenticator или OTP чрез SMS. Поради това, че криптографските алгоритми се изпълняват на заден фон, дължината на генерирания код не е от значение за бързината на действие, в резултат на което тя е значително удължена от приетите за норма 6 или 8 цифри.

Поради това, че именно на самото U2F устройство се съхранява частния ключ за криптиране, то на сървъра, удостоверяващ автентикацията

не е нужно съхраняването на таен низ, който може да бъде извлечен и използван отново.

U2F устройствата са изключително лесно достъпни от гледна точка на тяхното закупуване. Потребителите имат избор на доставчик, като цената на самите устройства варира между $6 и $50 в зависимост от производителите. За разлика от Token устройствата, често предоставяни от банки, U2F устройствата могат да се въведат в употреба от самия потребител и няма нужда институцията да настройва устройството.

Въпреки, че не е нужно инсталирането на специализиран софтуер за използването на U2F, технологията е достъпна само чрез съвременните версии на Google Chrome, Mozilla Firefox и Opera. Необходимо е и наличието на USB порт.

Една от най-сериозните разлики от другия вид двуфакторна автентикация обаче е именно вградената защита против Phishing и Man-In-The-Middle атаки. За коректната работа на U2F устройствата, сървърът записва т.нар. KeyHandle, в който са криптирани: адреса на сървъра, протоколът, по който се достъпва, както и порта на услугата. При опит за автентикация от грешен URL, алгоритъмът за проверка на валидността ще върне грешка при обработване, което на практика означава, че потребителят може да влиза в системата само от правилния URL. U2F устройствата работят при заявка от сървъра и ако комуникационните пакети се модифицират, то отговорът на устройството няма да съвпада с този, който сървърът очаква да получи. В резултат на това успешно влизане в системата би било невъзможно.

Ще разгледаме регистрирането на U2F устройство към акаунт, аналогично на разгледания по-горе пример за OTP.

```php
<?php
require_once('include/oauth/src/U2F/U2F.php');

// Инициране на U2F сървър класа
// URL за автентикация трябва да съвпада с посочения като аргумент
$test = new u2flib_server\U2F("https://webstudent.ue-varna.bg");

// Генерираме Challenge
$getRegisterData = $test->getRegisterData();

// Този сегмент се изпълнява само при натискане на хардуерния бутон на U2F
if($_POST){
    $request = json_decode($_SESSION['u2f']['getRegisterData']);
    $response = json_decode("[".$_POST['data']."]")[0];

    try {
        // На база Challenge-а се генерират keyHandle и publicKey,
        // които се съхраняват в базата
        $register_data = $test->doRegister($request, $response, $cert = true);

        $udata = array(
            'keyHandle' => $register_data->keyHandle,
            'publicKey' => $register_data->publicKey,
            'certificate' => $register_data->certificate,
            'counter' => $register_data->counter,
        );
        if(!$db->QueryUpdate("users", $udata, "id='".$_POST['user']."'")) {
            throw new Exception($MSG['MSG_ERROR_CHANGE_DATA']);
        }

        $skin->assign("SUCCESS", "Успех!");
    }
    catch(Exception $e) {
        // Принтиране на грешка
    }
}
// Съхраняваме Challenge-а във сесийна променлива,
// за да можем да я проверим при Submit на формуляра
$_SESSION['u2f']['getRegisterData'] = json_encode($getRegisterData[0]);

// Изпращаме генерирания от сървъра Challenge към JavaScript на клиента
$skin->assign("VARS", "var request = ".json_encode($getRegisterData[0]).";");
```

U2F технологията позволява регистрирането на повече от едно устройство към даден акаунт като по този начин дори потребителят да няма достъп до едно от устройствата, които притежава, да може да се автентикира успешно в системата.

Заключение

Използването на двуфакторна автентикация чрез методите, описани по-горе повишава нивото на сигурност на уеб приложенията, работещи в една несигурна и враждебна среда, каквато е Интернет. Макар приетите за стандарт методи за двуфакторна автентикация да имат слаби места, те несъмнено биха били в полза на потребителите, които желаят да притежават една по-висока степен на сигурност. Отвореният код на приложенията за имплементиране на технологията е предпоставка все повече системи да имплементират възможността за U2F автентикация. Универсалната двуфакторна автентикация (U2F) има потенциала да се превърне в нов стандарт поради огромните усилия на нейните разработчици и резултатите, които са постигнати.

Използвана литература

1. Дражев, С., Парушева, С., Петров, П. Стратегия за защита на уеб базирани банкови информационни системи. Сборник научни трудове: Посветен на 105-год. от рождението на пионерите на компютърната техника Джон Атанасов и Джон Фон Нойман, Шумен: Унив. изд. "Е. К. Преславски", 2, 2009, 29-35.
2. Brandom, Russell Two-factor authentication is a mess, 2017 <https://www.theverge.com/2017/7/10/15946642/two-factor-authentication-online-security-mess> (достъпено 01.05.2010г.)
3. Stanislav, Mark Two-Factor Authentication, IT Governance Publishing, 2015
4. Thatha, Rakesh Limitations of two factor authentication (2FA) technology, 2012 <https://www.computerweekly.com/tip/Limitations-of-two-factor-authentication-2FA-technology > (достъпено 01.05.2010г.)
5. Weiss, Kenneth Method and apparatus for positively identifying an individual, 1984 <https://patents.google.com/patent/US4720860> (достъпено 01.05.2010г.)