ONE TIME PASSWORDLESS and IP ADDRESS AUTHENTICATION METHOD for WEB APPLICATION

Mohammad Sani Suprayogi
Program Studi Teknik Informatika
Universitas Semarang
Semarang, Indonesia
yogie@usm.ac.id

Abstract - Penelitian yang membahas model autentikasi pengguna, mulai dari autentikasi tradisional menggunakan *username* dan *password* hingga metode *multi-factorauthentication*telah sering dilakukan. Namun model autentikasi tersebut masih menggunakan *password*, dimana manusia memiki keterbatasan untuk mengingat sehingga resiko kehilangan password kerap terjadi. Selain itu pencurian data pada jaringan komputer masih marak dilakukan sehingga perlu pendekatan lain dalam autentikasi pengguna terhadap sistem.

Passwordlessauthentication adalah model autentikasi yang mulai dikenalkan, hanya saja proses implementasinya masih terbatas. Paper ini berusaha meningkatkan metode *passwordless* dengan tambahan *time limit, session,* dan *ipaddress* dalam melakukan autentikasi pengguna.

Hasilnya, pengguna tidak perlu membuat dan mengingat *password*. Pengguna cukup memanfaatkan layanan email untuk proses registrasi dan login, kemudian *ipaddress* menjamin bahwa hanya pengguna tersebut yang dapat mengakses layanan website.

Keywords — passwordless; authentication; security;

I. PENDAHULUAN

Pada aplikasi web, faktor keamanan akan selalu menjadi isu utama.Aplikasi yang dibangun dengan bahasa PHP perlu memperhatikan lima ancaman dari peretas, yaitu[1]: *remote code execution, cross-site scripting (XSS), SQL injection (SQLI),* dan *PHP configuration and file system attacks*. Selain itu,autentikasi pengguna juga perlu diperhatikan supaya menjamin kenyamanan dan keamanan pengguna.

Penelitian berbagai model autentikasi pengguna pada aplikasi web telah sering dilakukan, namun pada sisi implementasi masih menggunakan model tradisional dengan pemanfaatan *username* dan *password* sebagai basis autentikasi *user*. Padahal Xiaoling and Jidong [1] menyebutkan bahwa penggunaan *username* dan *password* memiliki kelemahan, proses autentikasi yang menggunakan data yang tersimpan di server yang sama akan meningkatkan resiko keamanan meskipun telah menerapkan algoritma enkripsi seperti MD5.

Penelitian yang membahas *passwordless* sebelumnya telah dilakukan. Shirali-Shahreza and Shirali-Shahreza [2] telah menggunakan CAPTCHA[2] tetapi baru dikhususkan pada aplikasi mobile. Pembahasan untuk aplikasi web telah dilakukan namun masih mengandalkan *cookies web browser* sebagai pelindung[3]dimana*cookies* dapat diretas, selain itu ancaman penyadapan pada jaringan juga masih terbuka [4-6]. Sebelumnya Sælensminde and Boonjing [7] mengusulkan metode *passwordless* yang sederhana, maka kali ini penulis mempertimbangkan pemanfaatan *time limit, session,* dan *ipaddress* untuk meningkatkan penelitian tersebut.

Paper ini terstruktur sebagai berikut: Bagian II membahas sistem sekarang. Bagian III mengenai sistem yang diusulkan dan bagian IV merupakan contoh implementasi dan dibahas pada bab V. Kami memberikan kesimpulan pada bagian VI.

[1]https://www.owasp.org/index.php/PHP_Top_5

[2] Completely Automated Public Turing test to tell Computers and Human Apart

II. SISTEM SEKARANG

Proses registrasi dan *login* pada aplikasi web secara umum masih menerapkan metode *username* dan *password* sebagai autentikasi pengguna walaupun beberapa layanan telah meningkatkan pengamanannya dengan metode *multi-factorauthentication*. Internet Banking di Indonesia telah mewajibkan penggunaan perangkat token untuk mengakses fitur perbankan, Outlook dan WordPress memanfaatkan aplikasi OTP *generator*, kemudian Facebook dan Gmail telah menerapkan *two-factorauthentication* memanfaatkan teknologi SMS.

Penelitian mengenai peningkatan dari sistem autentikasi tradisional menjadi *multi-factorauthentication* juga telah dilakukan [8-12], namun pengguna tetap diharuskan untuk menginput*username* dan *password* untuk dapat menggunakan layanan. Selain itu pemanfaatan teknologi SMS memiliki kelemahan apabila pengguna tidak berada pada wilayah yang tercover sinyal telepon. Miller [13] mengatakan bahwa *two-stepverification* hanya cocok untuk *geek*, sedangkan bagi pengguna biasa proses autentikasi ini dirasa menyeramkan.

Graham [14]telah mengimplementasikan layanan tanpa *password* pada aplikasi Shop ItToMe[3], dimana pengguna cukup menggunakan akun email untuk mendaftar dan mengakses layanannya. Pada aplikasi tersebut, pengguna hanya mengisi alamat email pada *form* yang disediakan, kemudian sistem akan mengirimkan balasan email yang berisi alamat *url* unik yang hanya dapat diakses sekali waktu.

Metodeautentikasi yang penulisusulkan mengadopsi sistem autentikasi Shop ItToMe namun penggunaannya diperuntukkan ke aplikasi web dengan pengamanan yang lebih diperketat.

III. SISTEM YANG DIUSULKAN

Pada metode yang diusulkan, setiap pengguna akan melakukan beberapa hal berikut untuk mengakses web:

1. Pengguna klik menu *login*.
2. Pengguna memasukkan alamat email sebagai *username*.
3. Sistem memeriksa alamat email apakah telah terdaftar.

A. Email Belum Terdaftar

Apabila email belum terdaftar, maka sistem akan melakukan proses registrasi dengan langkah sebagai berikut:

1. Sistem mencatat *ipaddress* pengguna dan waktu akses kemudian mengirimkan email yang berisi alamat *url*aktivasi beserta informasi *time limit*.
2. Pengguna melakukan klik pada alamat *url*.
3. Sistem mengecek apabila pengguna telah melewati *time limit* maka pengguna diwajibkan mengulang proses registrasi.
4. Sistem melakukan autentikasi berdasarkan alamat *url* yang diakses beserta *ipaddress* yang tercatat.
5. Bila autentikasi berhasil, sistem akan membuatkan *session* bagi pengguna.
6. Pengguna telah terdaftar dan mendapatkan akses ke dalam website.

B. Email Telah Terdaftar

Apabila email telah terdaftar, maka sistem akan melakukan prosedur login dengan langkah sebagai berikut:

1. Sistem mencatat *ipaddress* pengguna dan waktu akses kemudian mengirimkan email berisi alamat *url* yang digunakan untuk mengakses dashboard beserta informasi *time limit*.
2. Pengguna melakukan klik pada alamat *url*.
3. Sistem mengecek apabila pengguna telah melewati *time limit* maka pengguna diwajibkan mengulang prosedur *login*.
4. Sistem melakukan autentikasi berdasarkan alamat *url* yang diakses beserta *ipaddress* yang tercatat.
5. Bila autentikasi berhasil, sistem akan membuatkan *session* bagi pengguna.
6. Pengguna mendapatkan akses ke dalam website.

[3]https://itunes.apple.com/us/app/shop-it-to-me-designer-clothes/id739426106

Pada metode yang diusulkan ini, pengguna tidak perlu membuat atau mengingat *password* untuk mengakses web. Pengguna hanya memasukkan alamat email sebagai *username* kemudian membuka *inbox* email untuk mengakses webmelalui alamat *url* unik yang dikirimkan oleh sistem. Kerahasiaan data diserahkan kepada penyedia layanan email yang telah memiliki sistem pengamanan data sendiri ditambah perlindungan dari SSL[4].

Selain itu pengguna juga diharuskan menggunakan perangkat yang sama untuk mengakses web karena sistem autentikasi bekerja berdasarkan *ipaddress* yang digunakan saat proses *login*, sehingga dapat mengurangi ancaman penyadapan alamat *url*.

IV. IMPLEMENTASI

Implementasi yang kami lakukan tidak mencakup keseluruhan sistem, namun membahas pada bagian utama autentikasi yang meliputi:

1. Pengecekan alamat email.
2. Pengiriman alamat url.
3. Autentikasi pengguna
4. Login ke website

A. *Pengecekan Alamat Email*

Fungsi utama pada tahap ini melakukan pengecekan terhadap alamat email apakah pengunjung telah terdaftar sebagai pengguna yang kemudian akan menentukan prosedur selanjutnya.

get email address
if email address is not registered
if email address format is valid
do registration
    end if
else
do login
end if

B. *Pengiriman Alamat URL*

Proses registrasi dan login memiliki fitur yang serupa yaitu mengirimkan email yang berisi alamat url sebagai jalan untuk mengakses website. Setiap url akan bersifat unik dan hanya dapat digunakan sekali waktu.

get ip address
get email address
get access time
set access time to H:i:s format

function token (access time, ip address) {
    create encryption from access time and ip address
}

function url (email address, token) {
    get domain name
    create url from domain name and token
}

function limit (access time) {
    addaccess time to 10 minutes
}

put ip address to database
put time access to database
put limit to database
put token to database
put url to database
put email address to database
send url to email address

C. *Autentikasi Pengguna*

Tahap autentikasi dilakukan setelah pengguna melakukan klik pada tautan *url* yang dikirimkan ke email. Sistem akan memeriksa *ipaddress* dan waktu akses pengguna dalam melakukan validasi.

get url
get ip address
getaccess time

function timing (access time) {
    get limit
    ifaccess time< limit
        timing is true
    else
        timing is false
    end if
}

if timing is true
    if ip address is same with database
        if url is same with database
            get email address
            if email address is registered
                do login
            else
                register email address
                do login
            end
        end if
    end if
end if

[4]Security Socket Layer

D. Login ke Website

Tahap ini adalah saat pengguna berhasil diautentikasi oleh sistem, selanjutnya sistem akan mengikat pengguna dengan *PHP session* supaya dapat menjelajahi halaman web.

```
if ip address is same with database
    if session is not registered
        register session
        access granted
    else
        access granted
    end if
end if
```

V. PEMBAHASAN SISTEM

Berdasarkan implementasi sederhana yang kami lakukan, pengguna tidak perlu membuat password untuk registrasi dan mengakses web. Sistem secara otomatis akan melakukan validasi alamat email, baik secara format penulisan ataupun status keanggotaan pada web. Selanjutnya, sistem akan mengirimkan email yang berisi alamat url unik yang terbuat dari kombinasi ipaddress dan waktu akses, sehingga alamat unik tersebut hanya dapat diakses oleh ipaddress tertentu dan dibatasi waktunya selama 10 menit setelah url terkirim.

Sistem autentikasi yang kami kerjakan dapat mengurangi resiko penyadapan yang dilakukan oleh peretas. Hal ini disebabkan alamat *url* yang bersifat unik hanya dapat diakses oleh *ipaddress* yang terdaftar, selain itu alamat *url* hanya dapat diakses dalam jangka waktu 10 menit setelah terkirim ke alamat email pengguna. Session yang tercipta saat pengguna berhasil *login* ke web juga telah terlindungi berdasarkan *ipaddress*.

VI. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian yang telah didapatkan, maka kami dapat menarik kesimpulan yaitu:

1. Metode *passwordless* dapat digunakan pada aplikasi web untuk mengurangi ketergantungan pengguna terhadap *password*.
2. Pengamanan berdasarkan pembatasan waktu akses, *session*, dan *ipaddress* dapat mengurangi resiko penyadapan oleh peretas.

Sebagai saran untuk penelitian selanjutnya, kami menyarankan hal-hal berikut:

1. Perlu penelitian lebih lanjut mengenai algoritma enkripsi yang digunakan saat pembuatan token.
2. Pengamanan lalu lintas data antara pengguna dengan *server* perlu diteliti lebih lanjut.

REFERENCES

[1] Z. Xiaoling and J. Jidong.,"*Research for the application and safety of MD5 algorithm in password authentication in Fuzzy Systems and Knowledge Discovery (FSKD)",*9th International Conference on 2012, pp. 2216-2219.

[2] M. Shirali-Shahreza and S. Shirali-Shahreza, "*Passwordless login system for mobile phones using CAPTCHA",* in *ELMAR 2007,* pp. 243-246.

[3] F. Heinemann and R. Nyman., "*Passwordless authentication: Secure, simple, and fast to deploy"* Available: https://hacks.mozilla.org/2014/10/passwordless-authentication-secure-simple-and-fast-to-deploy/

[4] S. Ansari, S. G. Rajeev, and H. S. Chandrashekar, *"Packet sniffing: a brief introduction",* Potentials 2002, *IEEE,* vol. 21, pp. 17-19.

[5] T. Chomsiri, "*Sniffing Packets on LAN without ARP Spoofing*. in *Convergence and Hybrid Information Technology",* ICCIT 2008, Third International Conference on 2008, pp. 472-477.

[6] S. Gupta and L. Sharma, "*Exploitation of Cross-Site Scripting (XSS) Vulnerability on Real World Web Applications and its Defense*", International Journal of Computer Applications, vol. 60, pp. 28-33.

[7] K. Sælensminde and V. Boonjing, "*Simple password less authentication system for web sites*", in Information

Technology: New Generations (ITNG), 2010 Seventh International Conference on, 2010, pp. 132-137.

[8] S. K. Joram, G. Pelechaty, P. K. Chauhan, and S. Vittal, *"Multiple factor user authentication system,"* US 2009/0063850 A1, Mar. 5, 2009.

[9] Y. Onda, S. SeongHan, K. Kobara, and H. Imai, *"How to distinguish on-line dictionary attacks and password mis-typing in two-factor authentication,"* in Information Theory and its Applications (ISITA), 2010 International Symposium on, 2010, pp. 571-576.

[10] J. Steinberg, *"System and method of using two or more multi-factor authentication mechanisms to authenticate online parties,"* US 2007/0136573 A1, Jun. 14, 2007.

[11] M. Ziqing, D. Florencio, and C. Herley, *"Painless migration from passwords to two factor authentication,"* in Information Forensics and Security (WIFS), 2011 IEEE International Workshop on, 2011, pp. 1-6.

[12] Dr.P.Pandarinath and V. Yalamanchili, *"Improved Password Authentication System against Password attacks for web Applications",* International Journal of Computer Trends and Technology (IJCTT), vol. 4, pp. 2878-2883, 2013.

[13] R. Miller, *"Why two-step verification will never work*", Available: http://www.citeworld.com/article/2115319/mobile-byod/two-step-verification-sucks.html.

[14] C. Graham, "*Bye Bye, Password*", Available: http://blog.shopittome.com/2014/05/29/bye-bye-passwords/