DETEKSI DAN MITIGASI SERANGAN *BACKDOOR* MENGGUNAKAN *PYTHON WATCHDOG*

[1]Susilo Hartono, [2]Hartono, [3]Khusnul Khotimah
[1]susilohartono@umpri.ac.id, [2]hartono@umko.ac.id , [3]khusnul.khotimah@umko.ac.id

[1]Universitas Muhammadiyah Pringsewu, [2,3]Universitas Muhammadiyah Kotabumi

Abstract: *The number of cyber attacks is increasing. This happens thoroughly, both at the international and national levels. Technology, techniques, and methods of carrying out cyber attacks are also increasingly sophisticated and up-to-date. Responding to this phenomenon, this research was conducted to implement an application for detecting and mitigating backdoor-based attacks using Python Watchdog. The method used in this study is the experimental method. This research is a case study of backdoor attacks that have been experienced by Universitas Muhammadiyah Kotabumi. In August – December 2021, one of the servers owned by Universitas Muhammadiyah Kotabumi received a significant backdoor-based attack. This research implements Python Watchdog to detect foreign files that are indicated as a backdoor, then sends notifications. Referring to the notification, the administrator can take further action. Based on the research that has been done, Python Watchdog is proven to overcome backdoor attacks. Once Python Watchdog is enabled, backdoor attacks are no longer possible.*

Keywords*: Cyber Security, Backdoor, Python, Watchdog, UMKO*

Abstrak: Jumlah serangan siber semakin meningkat. Hal ini terjadi secara menyeluruh, baik pada tingkatan internasional maupun nasional. Teknologi, teknik, dan metode melakukan serangan siber juga semakin canggih dan muktahir. Menyikapi fenomena tersebut, penelitian ini dilakukan untuk mengimplementasikan aplikasi deteksi dan mitigasi serangan berbasis *backdoor* menggunakan Python Watchdog. Metode yang digunakan pada penelitian ini adalah metode eksperimen. Penelitian merupakan studi kasus serangan *backdoor* yang pernah dialami oleh Universitas Muhammadiyah Kotabumi. Pada Agustus – Desember 2021, salah satu *server* milik Universitas Muhammadiyah Kotabumi mendapatkan serangan berbasis *backdoor* yang cukup signifikan. Penelitian ini mengimplementasikan Python Watchdog untuk mendeteksi berkas asing yang terindikasi *backdoor*, kemudian mengirimkan notifikasi. Mengacu notifikasi tersebut, administrator dapat mengambil tindakan lebih lanjut. Berdasarkan penelitian yang telah dilakukan, Python Watchdog terbukti mengatasi serangan *backdoor*. Setelah Python Watchdog diaktifkan, serangan *backdoor* tidak lagi dapat dilakukan.

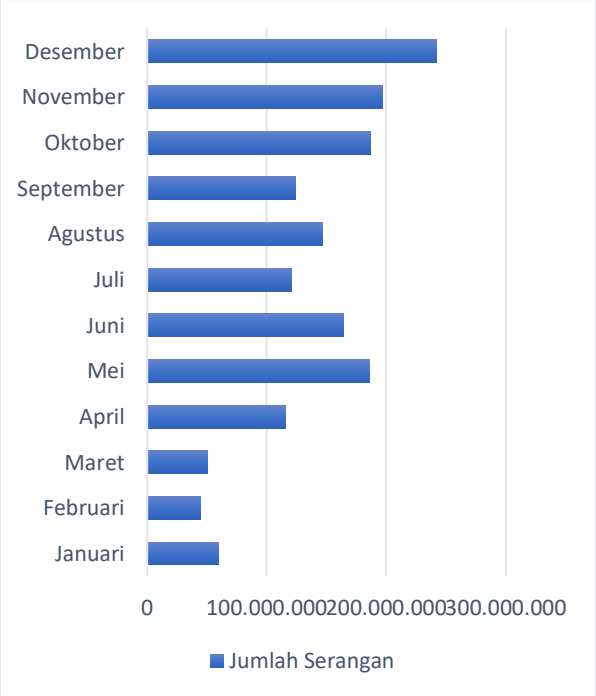
Kata kunci: Kemanan siber, *Backdoor, Python, Watchdog,* UMKO

I. PENDAHULUAN

Serangan siber merupakan upaya yang dilakukan oleh individu atau kelompok untuk mengakses atau mengganggu sistem komputer atau perangkat dalam jaringan. Serangan siber dapat berupa upaya mencuri data sensitif, merusak atau menghapus data, mencuri uang atau informasi keuangan, atau menciptakan gangguan pada sistem (Luthfah, 2021). Serangan siber dapat dilakukan oleh penjahat siber yang mencari keuntungan finansial, spionase industri atau keamanan nasional, atau aktivis politik atau sosial. Setiyawan dan Churniawan mengatakan bahwa sera-ngan

siber saat ini dapat dipandangan sebagai serangan non-militer karena dapat berlanjut pada upaya propaganda, provokasi, dan serangan informasi (Setiyawan et al., 2020). Menurut laporan tahunan Badan Sandi dan Siber Negara Republik Indonesia Tahun 2021, jumlah serangan anomali pada tahun 2021 mencapai 1.637.973.022 (Yusuf, 2022).

Serangan siber terjadi melalui berbagai cara, seperti *malware* atau virus komputer yang menyerang sistem, *phishing*, serangan *Denial of Service* (DoS), dan lain-lain yang bertujuan untuk membuat sistem menjadi tidak dapat diakses oleh pengguna. Serangan siber dapat memiliki dampak yang merugikan bagi perusahaan, organisasi, atau individu yang menjadi sasaran (Islami, 2018). Serangan siber dapat menyebabkan kehilangan data berharga, mengganggu operasi bisnis, dan merusak reputasi perusahaan a(Parulian et al., 2021). Bahkan serangan siber yang tampaknya kecil dan tidak signifikan dapat menjadi pintu masuk bagi penyerang untuk melakukan serangan lebih besar yang dapat menimbulkan kerugian besar (Hartono, 2022). Berikut ini adalah grafik anomali nasional pada tahun 2021 (grafik 1).



Grafik 1. Data Anomali Nasional Tahun 2021

Untuk menghindari serangan siber, perusahaan, organisasi, dan individu harus mampu mengambil langkah-langkah keamanan siber yang tepat. Terdapat banyak metode dan teknik yang dapat digunakan untuk melakukan serangan siber seperti *Cross Site Scripting* (XSS), *Remote Code Execution* (RCE), *Insercure Direct Object Reference* (IDOR), dan lain-lain. Dalam kasus di Indonesia, salah satu metode dan teknik serangan yang paling sering ditemukan adalah RCE dengan bantuan *backdoor shell* (Yusuf, 2022).

Setelah penyerang menanamkan *backdoor*, penyerang dapat melakukan serangan secara mudah. Bahkan, penyerang dapat memiliki akses super admin dan mengambil semua sumber daya server (Anusha, 2020; Biswas, 2018). Apabila server mengelola atau menyimpan data penting dan sensitif, tentunya *backdoor* menjadi masalah yang sangat serius. Mudahnya akses untuk mendapatkan berbagai *backdoor script/shell* semakin meningkatkan keberhasilan atau peluang penyerang dalam menanamkan *backdoor*. Bahkan, pada beberapa kasus, penyerang golongan *kiddies* mampu melakukan serangan yang cukup kronis ketika menggunakan bantuan *backdoor*.

Ketika penyerang berhasil menanamkan *backdoor* ke dalam suatu server, aturan dan pembatasan perilaku yang telah ditetapkan tidak dapat diberlakukan secara efektif (Tan & Soewito, 2022). Penyerang dapat bekerja dan bertindak sebagai *root* yang artinya pemilik utama *server*. Penyerang dapat bebas mengakses seluruh sumber daya *server* mulai dari berbagai berkas, folder, dan database tanpa adanya pembatasan yang berarti (Sureda Riera et al., 2020). Sejalan dengan maknanya, penyerang dapat keluar masuk melalui pintu belakang (*backdoor*) secara bebas kapanpun mereka menginginkannya. Dengan demikian, *backdoor* merupakan suatu ancaman serius, yang merupakan awal dari serangan-serangan lanjutan yang lebih kompleks dan berbahaya.

Tabel 1. Kasus Serangan Web-Defacement Di Indonesia Tahun 2021

No	Bulan	Jumlah *Defacement*
1	Januari	419
2	Februari	398
3	Maret	727
4	April	526
5	Mei	453
6	Juni	656
7	Juli	482
8	Agustus	427
9	September	326
10	Oktober	436
11	November	518
12	Desember	572

Pada kenyataannya, meskipun *backdoor* adalah sesuatu yang sangat berbahaya dan dapat mengancam stabilitas sistem dan kerahasiaan data, kasus-kasus serangan yang memanfaatkan *backdoor* masih banyak ditemukan, terutama di Indonesia. Seperti yang tertera pada tabel 1, kasus serangan *website deface* yang sering dialami oleh banyak laman di Indonesia, mayoritas menggunakan bantuan *backdoor* untuk mengunggah dokumen *deface*-nya (Hasibuan & Gultom, 2018). Salah satu penyebab masih banyak ditemukannya serangan berbasis *backdoor* adalah sulitnya melakukan deteksi pada serangan ini. Administrator sistem tidak secara mudah memastikan apakah berkas tersebut *backdoor* atau bukan.

Salah satu server milik Universitas Muhammadiyah Kotabumi (UMKO) tercatat pernah mendapatkan serangan RCE berbasis *backdoor*. Informasi serangan berbasis *backdoor* tersebut dikirimkan oleh aplikasi imunifyAV yang merupakan salah satu fitur aplikasi cPanel. Berdasar pada data *history*, serangan berbasis *backdoor* pertama kali terdeteksi oleh sistem pada bulan Agustus 2021. Setelah itu, serangan-serangan berikutnya semakin meningkat. Memasuki bulan September 2021, serangan berbasis *backdoor* semakin meningkat. Berdasarkan data *history*, jumlah berkas yang terdeteksi sebagai *backdoor* mencapai 15 berkas. Hal ini menunjukan bahwa terdapat celah keamanan yang tereksploitasi dan aktivitas ilegal meningkat. Serangan berbasis *backdoor* dapat menjadi pintu awal bagi serangan-serangan lanjutan.

Terdapat beberapa penelitian yang pernah membahas dan mengaji tentang *backdoor*. Kebanyakan tujuan penelitian tersebut adalah untuk mendeteksi keberadaan berkas *backdoor* di server, karena dapat digunakan untuk melakukan serangan RCE. Penelitian Sopaheluwakan dan Chandra membahas tentang *web shell backdoor* berbasis bahasa Pemrograman PHP (Sopaheluwakan & Chandra, 2020). Sejalan dengan penelitian tersebut, penelitian terkait *monitoring* serangan berbasis *website* juga pernah dilakukan untuk mendeteksi serangan *backdoor* (Gumilang & Chandra, 2021). Metode *signature-based* dan *static analysis* juga pernah dignakan untuk menganalisa dan mendeteksi keberadaan *backdoor*, terutama pada kasus *content management system* (Hariyadi et al., 2022).

Secara umum, tujuan penelitian ini adalah untuk melakukan eksperimen *cyber* security. Secara khusus, tujuan penelitian ini adalah mengimplementasikan Python Watchdog untuk mendeteksi keberadaan *backdoor*. Setelah berkas terindikasi *backdoor* terdeteksi, sistem akan mengirimkan notifikasi ke email administrator sehingga dapat diambil langkah selanjutnya. Penelitian ini juga akan mengukur bagaimana efektivitas penerapan Python Watchdog dalam mengatasi serangan RCE berbasis *backdoor* di server UMKO.

II. SERANGAN BERBASIS *BACKDOOR*

Serangan backdoor adalah jenis serangan siber yang bertujuan mendapatkan akses tidak sah ke server atau sistem dengan memasang program yang disebut *shell* pada sistem. *Shell* merupakan program yang dapat

memungkinkan seseorang untuk memasukkan perintah pada sistem seperti yang dapat dilakukan melalui terminal atau *command prompt*. Sebuah *backdoor* merupakan program yang dipasang secara tersembunyi dan tanpa izin pada sistem sehingga penyerang dapat mengakses sistem tersebut dari jarak jauh tanpa sepengetahuan pengguna (Yu et al., 2021). *Backdoor* dipasang pada server dengan tujuan mencuri data sensitif seperti nama pengguna dan kata sandi, atau bahkan memodifikasi atau menghapus data yang ada pada server. *Backdoor* juga dapat digunakan untuk menginstal program jahat lainnya seperti malware, virus, dan trojan horse pada sistem (Zakaria, n.d.).

Backdoor dapat dipasang pada sistem melalui berbagai cara, termasuk melalui kerentanan pada perangkat lunak yang digunakan pada sistem atau melalui serangan phishing yang mengelabui pengguna untuk memasukkan informasi sensitif atau mengklik tautan yang membawa ke halaman palsu yang menginstal shell pada sistem. Untuk mencegah serangan backdoor shell, perlu dilakukan tindakan-tindakan keamanan yang tepat. Hal ini termasuk memastikan bahwa sistem dan perangkat lunak yang digunakan selalu diperbarui dengan versi terbaru yang memiliki *patch* keamanan terbaru dan mengatur kebijakan keamanan ketat pada *web server*.

Dalam rangka melindungi diri dari serangan backdoor shell, diperlukan pemahaman yang baik tentang ancaman keamanan siber dan tindakan yang dapat dilakukan untuk melindungi sistem dan data sensitif. Selalu pastikan untuk memperbarui sistem keamanan, menghindari praktik-praktik yang dapat membuka celah bagi serangan, dan menggunakan alat keamanan siber yang tepat untuk melindungi sistem dan data sensitif.

III. PYTHON WATCHDOG

Python Watchdog adalah sebuah pustaka Python yang memungkinkan Anda untuk memantau perubahan file dan direktori pada sistem operasi secara *real-time*. Watchdog sangat berguna untuk memantau berkas sistem. Dengan demikian, administrator dapat membuat suatu tindakan setelah adanya perubahan pada berkas atau direktori yang mencurigakan. Pustaka Python Watchdog dapat digunakan pada sistem operasi Windows, Mac OS X, dan Linux.

Pustaka ini memungkinkan pengembang untuk memantau berkas dan direktori seperti penambahan, penghapusan, dan perubahan berkas. Watchdog menyediakan sebuah API yang sangat intuitif dan mudah digunakan. Script Watchdog merupakan sebuah kelas yang mewarisi kelas FileSystemEventHandle. Setelah itu, Watchdog akan memantau direktori dan memanggil metode yang sesuai pada objek yang di daftarkan pada setiap kali terjadi perubahan pada berkas atau direktori yang dipantau. Berikut adalah contoh penggunaan Watchdog untuk memantau direktori dan mencetak pesan pada setiap kali terjadi perubahan pada direktori tersebut:

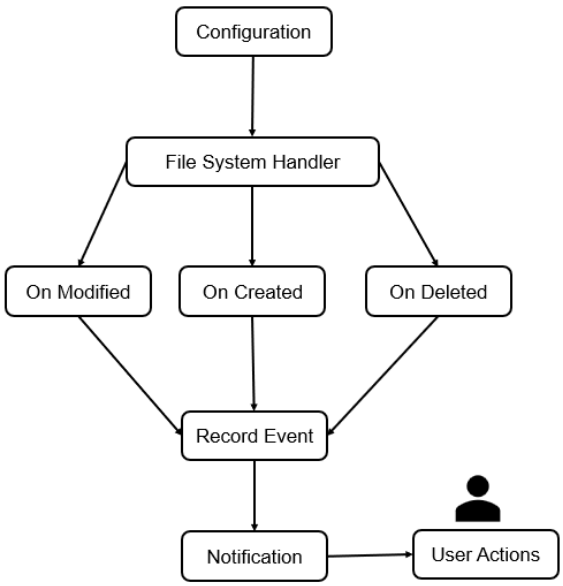
```
from watchdog.observers import Observer
from watchdog.events import FileSystemEventHandler

class Handler(FileSystemEventHandler):
    def on_modified(self, event):
        print(f"File {event.src_path} modified.")

if __name__ == "__main__":
    event_handler = MyHandler()
    observer = Observer()
    observer.schedule(event_handler, path='.')
    observer.start()
    try:
        while True:
            pass
    except KeyboardInterrupt:
        observer.stop()
    observer.join()
```

Pada contoh tersebut, kelas Handler mewarisi kelas FileSystemEventHandler dan mengimplementasikan metode *on_modified* untuk menangani perubahan berkas. Kemudian, objek Handler didaftarkan dengan *Observer* dan diberikan direktori yang dipantau. Setelah itu, *Observer* dimulai dan akan terus memantau direktori tersebut. Pada

penelitian ini, peneliti telah membuat *script* Watchdog yang dapat memantau ketika terjadi perubahan pada berkas dan direktori, terutama pada *event* saat dimodifikasi, dibuat, dan dihapus. Contoh di atas hanyalah gambaran bagaimana mengimplementasikan Python Watchdog dengan cara sederhana.

IV. METODE

Metode penelitian yang digunakan adalah metode eksperimen. Tujuan utama penelitian ini adalah mendeteksi berkas-berkas asing yang kemungkinan merupakan *backdoor* dan mela-kukan mitigasi untuk mencegah serangan berbahaya lebih lanjut. Untuk memastikan bahwa berkas asing yang dibuat tersebut bukanlah berkas yang diproses oleh sistem, *server*, atau aplikasi terotorisasi, *script watchdog* juga telah membuat daftar *whitelist* direktori sehingga hanya mendeteksi berkas-berkas asing pada direktori yang telah ditentukan. Secara rinci, metode yang diterapkan dapat dilihat pada gambar 1 berikut.



Gambar 1. Metode Deteksi dan Pencegahan Backdoor Menggunakan Python Watchdog

Metode eksperimen ini diterapkan pada 35 aplikasi *website* UMKO yang berada pada salah satu *web server* VPS UMKO yang menggunakan cPanel. Dengan kata lain, Python Watchdog memantau sebanyak 35 aplikasi berbasis *website* yang terdiri dari beragam teknologi, *platform*, atau CMS (*Content Management System*). Penjelasan dari masing-masing tahapan adalah sebagai berikut:

1. ***Configuration*:** tahapan melakukan inisialisasi atau membaca konfigurasi sehingga Python Watchdog dapat bekerja berdasarkan *path, target files, target extentions, ignored folders and files,* dan parameter konfigurasi lainnya.
2. ***FilesystemEventHandler*:** tahapan untuk mempersiapkan *event handler*, sehingga *Watchdog* dapat melakukan pemantauan berkas dan direktori.
3. ***On Modified, On Created*, dan *On Deleted*:** tahapan mendeteksi perubahan tergantung dari *event* yaitu modifikasi, pembuatan, dan penghapusan berkas.
4. ***Record Event*:** tahapan untuk merekam, mendeteksi, dan mengi-rimkan informasi perubahan sehing-ga menjadi *trigger* untuk mengaktif-kan notifikasi.
5. ***Notification*:** tahapan mengirimkan notifikasi ke administrator, sehingga dapat mengambil langkah-langkah

UMKO menggunakan layanan VPS berbasis Linux CentOS. Rincian spesifikasi VPS dan *webserver* yang digunakan sebagai berikut.

Sistem Operasi : Linux CentOS
RAM : 8 Gb
Penyimpanan : 120 Gb
PHP Version : 7.4
Apache Version : 2.4.55
MySQL Version : 5.7.41
Arsitektur : X86_64

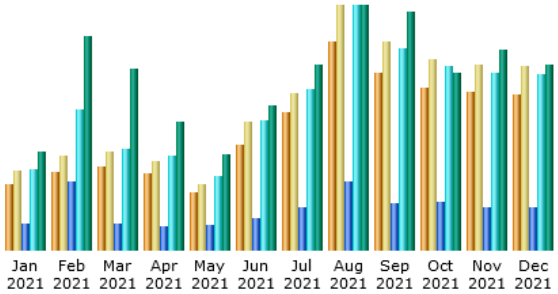
V. HASIL DAN PEMBAHASAN

1. Hasil

a) Statistik Website

Selama masa penelitian, jumlah pengunjung, halaman dikunjungi, *hit*, dan *bandwith* yang dikeluarkan oleh VPS UMKO

cukup besar. Dari besarnya jumlah tersebut, sistem VPS/cPanel UMKO belum mampu memisahkan statistik serangan dan non-serangan. Namun, statistik ini dapat menjadi gambaran terkait bagaimana lalu lintas keluar masuk selama penelitian. Lalu lintas kunjungan bulanan menuju laman utama UMKO digambarkan pada gambar 2 berikut ini.



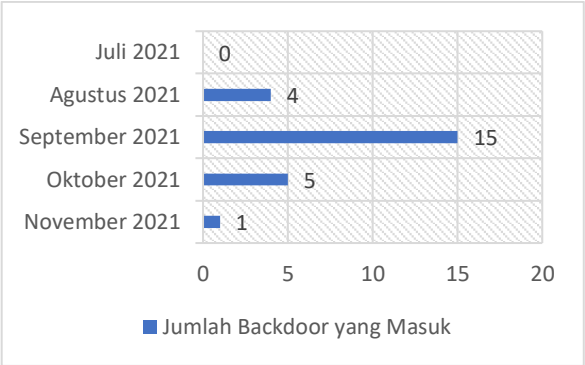
Gambar 2. Lalu Lintas Kunjungan ke https://www.umko.ac.id

Tabel 2. Rincian Lalu Lintas Kunjungan

Month	Unique visitors	Number of visits	Pages	Hits	Bandwidth
Jan 2021	9,189	11,045	95,405	286,232	40.85 GB
Feb 2021	10,906	13,140	243,585	497,765	88.39 GB
Mar 2021	11,625	13,722	95,627	358,875	75.19 GB
Apr 2021	10,688	12,354	85,160	335,045	53.31 GB
May 2021	7,972	9,165	86,292	263,301	39.78 GB
Jun 2021	14,769	17,858	112,628	457,503	59.78 GB
Jul 2021	19,165	21,938	150,657	569,787	76.50 GB
Aug 2021	29,094	34,138	241,062	861,990	101.03 GB
Sep 2021	24,662	29,025	166,041	712,313	98.54 GB
Oct 2021	22,643	26,604	168,698	649,240	73.45 GB
Nov 2021	22,185	25,939	149,374	627,012	82.77 GB
Dec 2021	21,629	25,685	149,140	621,616	76.44 GB
Total	204,527	240,613	1,743,669	6,240,679	866.04 GB

b) Serangan Sebelum Implementasi

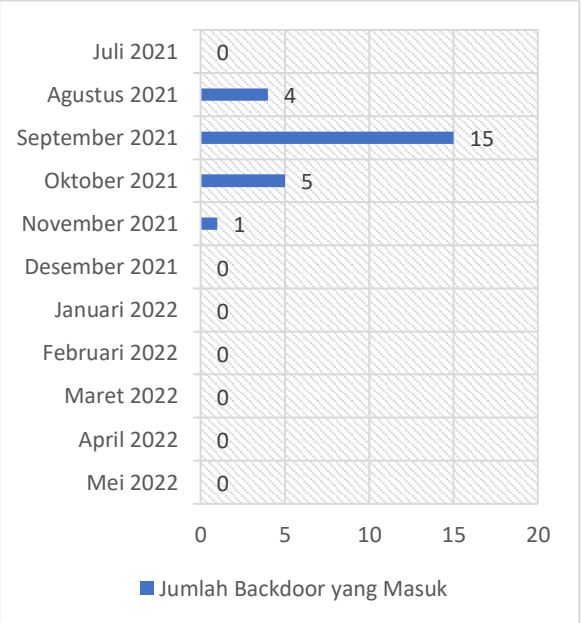
Serangan berbasis *backdoor* ke VPS UMKO pertama kali terdeteksi pada Agustus 2021. Pada masa tersebut, jumlah serangan *backdoor* yang masuk mencapai 4 serangan. Kemudian, pada September 2021, serangan *backdoor* meningkat signifikan, lebih tiga kali lipatnya, yaitu mencapai 15 serangan. Sementara itu, pada bulan Oktober 2021 terdapat 5 serangan dan November 2021 sebanyak 1 serangan berbasis *backdoor*. Semua serangan baru disadari pada awal November 2021, dan pada awal bulan tersebut aplikasi deteksi dan mitigasi serangan *backdoor* menggunakan pustaka Python Watchdog mulai diimplementasikan. Jumlah serangan *backdoor* yang masuk dilihat pada grafik 2 berikut ini.



Grafik 2. Serangan *Backdoor* Sebelum Implementasi

c) Serangan Sesudah Implementasi

Serangan *backdoor* baru diketahui administrator web pada November 2021. Untuk mengatasi serangan lanjutan, aplikasi deteksi dan mitigasi serangan *backdoor* menggunakan Python Watchdog diimplementasikan dan diujicobakan. Berdasarkan catatan atau log serangan, jumlah serangan *backdoor* pada bulan-bulan berikutnya berhasil diatasi, tepatnya pada Desember 2021 s.d Mei 2022. Dalam hal ini, tidak ada serangan *backdoor* yang dapat kembali masuk. Statistik serangan *backdoor* setelah diimplementasikan dapat dilihat pada grafik 3 berikut ini.



Grafik 3. Serangan *Backdoor* Sebelum Implementasi

Untuk mendapatkan data yang lebih rinci dan komprehensif, berkas, *cause*, dan *initiator* serangan *backdoor* dapat dilihat pada tabel 3

berikut ini. *Cause* adalah cakupan *backdoor* bekerja, dan *initiator* adalah pengguna yang membuat berkas tersebut. Ketika *initiator* adalah *root*, artinya penyerang dapat bertindak sebagai super administrator.

Tabel 3. Rincian Serangan *Backdoor* ke Server

Path	Cause	Initiator
/../../../wp-load.php	background	root
/../../../settings.php	background	root
/../../../gecko.php	background	root
/../../../wp-load.php	background	root
/../../../ahu.php	background	root
/../../../gecko.php	background	root
/../../../txtesiLur	user	root
/../../../fx.php	user	root
/../../../.htaccess	user	root
/../../../njir.php	user	root
/../../../wp-sorong-MAR.phtml	user	root
/../../../62e2a8fd9d6a0.phtml	user	root
/../../../62e2a67479859.phtml	user	root
/../../../txtesiLur	background	root
/../../../fx.php	background	root
/../../../.htaccess	background	root
/../../../njir.php	background	root
/../../../wp-sorong-MAR.phtml	background	root
/../../../62e2a8fd9d6a0.phtml	background	root
/../../../62e2a67479859.phtml	background	root
/../../../txtesiLur	background	root
/../../../wp-sorong-MAR.phtml	background	root
/../../../62e2a8fd9d6a0.phtml	background	root
/../../../62e2a67479859.phtml	background	root

d) Aplikasi dan CMS

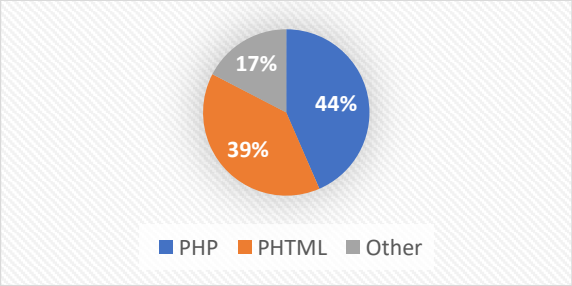
Aplikasi atau teknologi website dan CMS yang digunakan oleh UMKO sebenarnya beragam. Namun, serangan *backdoor* hanya memasuki beberapa jenis aplikasi dan CMS, yang kesemuanya berbasis bahasa pemrograman PHP. Daftar aplikasi dan CMS yang terinfeksi *backdoor* dapat dilihat pada tabel 4.

Tabel 4. Rincian Aplikasi/CMS Terinfeksi *Backdoor*

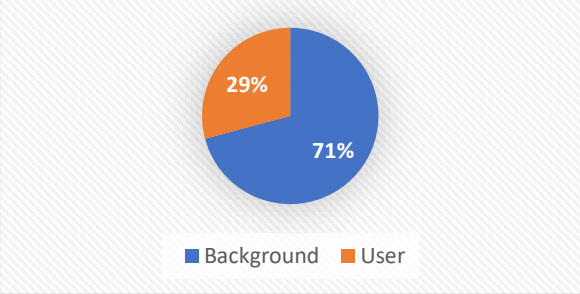
No	Aplikasi/CMS	Berkas Terinfeksi	Jumlah *Backdoor*
1	Wordpress	wp-load.php ahu.php gecko.php	4
2	Moodle	settings.php, gecko.php	1
3	Open Journal System	wp-sorong-MAR.phtml 62e2a8fd9d6a0.phtml 62e2a67479859.phtml txtesiLur	12
4	JDIH	fx.php .htaccess njir.php	6

e) Ekstensi dan *Cause*

Seperti yang telah diungkapkan bahwa serangan *backdoor* pada kasus UMKO hanya berpengaruh pada aplikasi berbasis bahasa pemrograman PHP. Dengan kata lain, ekstensi PHP dan PHTML menjadi yang paling dominan. Porsi ekstensi terinfeksi *backdoor* dan *cause* serangan *backdoor* dapat dilihat pada grafik 5 dan 6 berikut ini.



Grafik 4. Grafik Porsi Ekstensi Terinfeksi *Backdoor*



Grafik 5. Grafik Porsi *Cause* Serangan *Backdoor*

2. Pembahasan

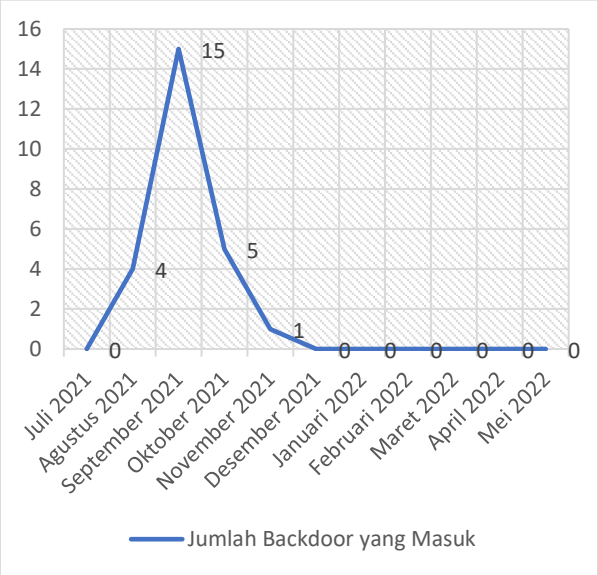
a) Tindakan Mitigasi

Pustaka *Python Watchdog* tidak hanya digunakan untuk mendeteksi berkas yang baru dibuat, dimodifikasi, dan dihapus, tetapi juga untuk mendeteksi apakah berkas tersebut terindikasi *backdoor shell* atau bukan. Setelah data notifikasi dikirimkan ke administrator *website*, tindakan lanjutan dapat diambil

apakah akan memasukan berkas tersebut pada daftar *white list* atau menghapus berkas tersebut. Tindakan mitigasi ini memang tidak dilakukan secara otomatis karena dikhawatirkan Python Watchdog mendeteksi berkas yang memang dilakukan secara sah oleh sistem, bukan penyerang .

b) Efektivitas Implementasi

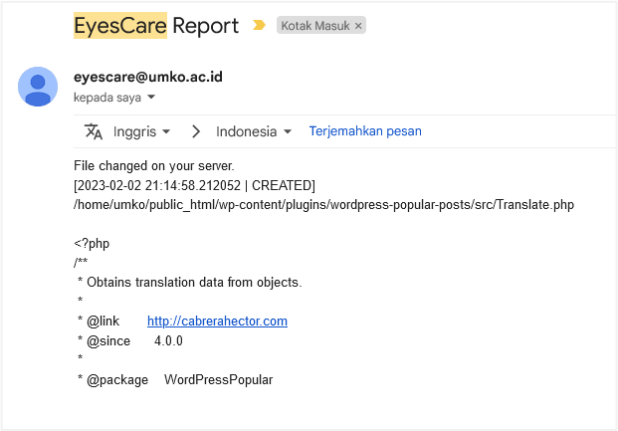
Berdasarkan hasil eksperimen, implementasi Python Watchdog terbukti efektif. Python Watchdog berhasil mendeteksi dan mengirim notifikasi ketika terdapat berkas yang mencurigakan. Hal ini semakin mempermudah administrator *website* untuk mengambil tindakan mitigasi lanjutan. Dinamika lalu lintas serangan *backdoor* setelah diimplementasikan dapat dilihat pada grafik 6.



Grafik 6. Serangan *Backdoor* Sebelum Implementasi

c) Notifikasi

Python Watchdog mengirimkan notifikasi ke email administrator *website*. Data notifikasi berisi informasi *event type*, *event datetime*, dan konten *head* berkas yang terdeteksi. Melalui informasi tersebut, administrator *website* dapat mengambil langkah mitigasi lanjutan sehingga serangan lanjutan dapat diatasi. Gambar 3 menunjukan salah satu notifikasi yang dikirimkan kepada administrator.



Gambar 3. Notifikasi Python Watchdog kepada Administrator

VI. KESIMPULAN

Berdasarkan serangkaian penelitian yang telah dilakukan, terdapat lima hal penting pada penelitian ini, yaitu sebagai berikut.

1. Python Watchdog mampu mendeteksi perubahan pada berkas secara cepat, mulai dari *on created*, *on modified*, dan *on deleted* dan mempermudah proses deteksi berkas yang terindikasi *backdoor*.
2. Implementasi Python Watchdog untuk mendeteksi dan melakukan mitigasi serangan backdoor pada kasus VPS UMKO terbukti efektif. Serangan backdoor tidak dapat dilakukan atau 0 sampai dengan bulan penelitian ini dilakukan.
3. Implementasi Python Watchdog terbukti dapat menurunkan intensitas serangan siber yang diarahkan pada VPS UMKO.
4. Meskipun pada tahap konfigurasi, terdapat pengaturan direktori atau berkas white list (tidak diindek), terdapat berkas atau direktori yang terbilang aman namun masih dianggap sebagai berkas yang mencurigakan sehingga masih memerlukan peran administrator untuk mengambil langkah selanjutnya.
5. Penelitian sejenis selanjutnya sebaiknya dilengkapi dengan deteksi berkas backdoor menggunakan machine learning agar proses deteksi dapat lebih tepat.

DAFTAR REFERENSI

Anusha, Z. F. (2020). Automatic Verification of a Remote Code Execution Vulnerability Detection Model Using the SPIN Model Checker. 35.

Biswas, S. (2018). A Study on Remote Code Execution Vulnerability in Web Applications. 8.

Gumilang, P. M. R., & Chandra, D. W. (2021). Implementasi dan modifikasi WebShell untuk monitoring serangan berbasis website. AITI, 18(1), 54–68. https://doi.org/10.24246/aiti.v18i1.54-68

Hariyadi, D., Setiawan, C. B., Sahtyawan, R., Wicaksono, A. I., & Wisnuaji, A. (2022). Analisis dan Deteksi Backdoor pada Content Management System Menggunakan Metode Signature-based dan Static Analysis. INTEK : Jurnal Informatika Dan Teknologi Informasi, 5(1), Article 1. https://doi.org/10.37729/intek.v5i1.1734

Hartono, O. W. P. (2022). Membangun dan Menguji Keamanan Website (1st ed.). Andi Publisher.

Hasibuan, M. S., & Gultom, L. M. (2018). Analisis Serangan Deface Menggunakan Backdoor Shell Pada Website. Techno.Com, 17(4), 415–423. https://doi.org/10.33633/tc.v17i4.1887

Islami, M. J. (2018). TANTANGAN DALAM IMPLEMENTASI STRATEGI KEAMANAN SIBER NASIONAL INDONESIA DITINJAU DARI PENILAIAN GLOBAL CYBERSECURITY INDEX. Masyarakat Telematika Dan Informasi : Jurnal Penelitian Teknologi Informasi dan Komunikasi, 8(2), 137. https://doi.org/10.17933/mti.v8i2.108

Luthfah, D. (2021). Serangan Siber Sebagai Penggunaan Kekuatan Bersenjata dalam Perspektif Hukum Keamanan Nasional Indonesia (Cyber Attacks as the Use of Force in the Perspective of Indonesia National Security Law). TerAs Law Review : Jurnal Hukum Humaniter Dan HAM, 3(1), 11–22. https://doi.org/10.25105/teras-lrev.v3i1.10742

Parulian, S., Pratiwi, D. A., & Yustina, M. C. (2021). Studi Tentang Ancaman dan Solusi Serangan Siber di Indonesia. Telecommunications, Networks, Electronics, and Computer Technologies (TELNECT), 1(2), Article 2.

Setiyawan, W. B. M., Churniawan, E., & Faried, F. S. (2020). UPAYA REGULASI TEKNOLOGI INFORMASI DALAM MENGHADAPI SERANGAN SIBER GUNA MENJAGA KEDAULATAN NEGARA KESATUAN REPUBLIK INDONESIA. 3(2).

Sopaheluwakan, C. R., & Chandra, D. W. (2020). Anti-WebShell PHP Backdoor Scanner pada Linux Server. ILKOM Jurnal Ilmiah, 12(2), 143–153. https://doi.org/10.33096/ilkom.v12i2.596.143-153

Sureda Riera, T., Bermejo Higuera, J.-R., Bermejo Higuera, J., Martínez Herraiz, J.-J., & Sicilia Montalvo, J.-A. (2020). Prevention and Fighting against Web Attacks through Anomaly Detection Technology. A Systematic Review. Sustainability, 12(12), Article 12. https://doi.org/10.3390/su12124945

Tan, T., & Soewito, B. (2022). Manajemen Risiko Serangan Siber Menggunakan Framework NIST Cybersecurity. JISAMAR (Journal of Information System, Applied, Management, Accounting and Research), 6(2), Article 2. https://doi.org/10.52362/jisamar.v6i2.781

Yu, X., Meng, W., Zhao, L., & Liu, Y. (2021). TridentShell: A Covert and Scalable Backdoor Injection Attack on Web Applications. In J. K. Liu, S. Katsikas, W. Meng, W. Susilo, & R. Intan (Eds.), Information Security (pp. 177–194). Springer International Publishing. https://doi.org/10.1007/978-3-030-91356-4_10

Yusuf, A. (2022). Laporan Tahunan 2020 Honeynet Project BSSN - IHP. Badan Siber dan Sandi Negara.

Zakaria, K. (n.d.). Mendeteksi Backdoor Dengan Aplikasi Shell Detektor. Kristison Zakaria, S.Pd.