

Forum:[Hack Forums](#)

Thread: [Does reCAPTCHA prevent brute force?](#)

Kriptik - March 31, 2021, 1:10 pm

Hey guys,

I'm working on a project and I'm already implementing some security features and I was thinking about a way on preventing attackers to brute force the login page, so I thought about reCAPTCHA.

But is that enough or should I do it another way?

BSSEC - March 31, 2021, 8:23 pm

(Mar 31, 2021 01:10 PM)Kriptik Wrote: [\[CITE\]](#) Hey guys,

I'm working on a project and I'm already implementing some security features and I was thinking about a way on preventing attackers to brute force the login page, so I thought about reCAPTCHA.

But is that enough or should I do it another way?

yeh most likely it kills the brute-force but it can be bypassed " if your project hold valuable informations i would suggest you to active a WAF a long side with recaptcha and you should also block ip address after 3 false logins this will make your project pain in the ass to brute-force it

good luck mate

Kriptik - April 1, 2021, 2:06 am

(Mar 31, 2021 08:23 PM)BSSEC Wrote: [\[CITE\]](#) yeh most likely it kills the brute-force but it can be bypassed " if your project hold valuable informations i would suggest you to active a WAF a long side with recaptcha and you should also block ip address after 3 false logins this will make your project pain in the ass to brute-force it

good luck mate

Thanks for the input, although I think blocking ip's is not really a good solution because an attacker could use proxies to go around it. Another idea would be to lock the actual account and maybe send an e-mail notification to the owner about the attempt. But for this I will have to look into the documentation on how to implement this feature.

Btw, how can reCAPTCHA be bypassed? Just curios.

Captain Pre - April 1, 2021, 2:10 am

(Apr 1, 2021 02:06 AM)Kriptik Wrote: [\[CITE\]](#) Thanks for the input, although I think blocking ip's is not really a good solution because an attacker could use proxies to go around it. Another idea would be to lock the actual account and maybe send an e-mail notification to the owner about the attempt. But for this I will have to look into the documentation on how to implement this feature.

Btw, how can reCAPTCHA be bypassed? Just curios.

There are captcha solving services, It resolves the captcha within 30 secs max.

The email idea is much better, Or just put up a 30mins waiting time after 3 failed login attempts (Get creative look at other websites)

[Found this from cloudflare if its helps](#)

BSSEC - April 1, 2021, 2:13 am

(Apr 1, 2021 02:10 AM)Captain Pre Wrote: [\[CITE\]](#) There are captcha solving services, It resolves the captcha within 30 secs max.

The email idea is much better, Or just put up a 30mins waiting time after 3 failed login attempts (Get creative look at other websites)

[Found this from cloudflare if its helps](#)

totally agree with this

Kriptik - April 1, 2021, 2:16 am

(Apr 1, 2021 02:10 AM)Captain Pre Wrote: [\[CITE\]](#) There are captcha solving services, It resolves the captcha within 30 secs max.

The email idea is much better, Or just put up a 30mins waiting time after 3 failed login attempts (Get creative look at other websites)

[Found this from cloudflare if its helps](#)

Thanks for your input. Cloudflare Rate Limiting seems like a nice service to have, I will definitely consider this option.

Th3Gh0st_Ma - April 1, 2021, 5:05 am

yes it stops the brute-force but it can be bypassed

Shadow_Cloud - April 1, 2021, 5:44 am

(Mar 31, 2021 01:10 PM)Kriptik Wrote: [\[CITE\]](#) Hey guys,

I'm working on a project and I'm already implementing some security features and I was thinking about a way on preventing attackers to brute force the login page, so I thought about reCAPTCHA.

But is that enough or should I do it another way?

You could also send a verification email for logins from a new IP block. So even if someone gets in via either a bf or by a user having the same credentials in a dump somewhere, the 2 factor auth will still protect the account?

With account lockout a spray attack is probably a bigger problem than brute force attacks. If you lock the account until after a password reset after x invalid login attempts, brute force is less likely to be an issue. You could also slow down any brute force attempt drastically by injecting a XSRF token on the login form and only processing logins from valid tokens.

There's a different version of IP 2fa called impossible travel that might be worth considering:

Spoiler ([Click to View](#))

This detection identifies two user activities (is a single or multiple sessions) originating from geographically distant locations within a time period shorter than the time it would have taken the user to travel from the first location to the second, indicating that a different user is using the same credentials. This detection uses a machine learning algorithm that ignores obvious "false positives" contributing to the impossible travel condition, such as VPNs and locations regularly used by other users in the organization. The detection has an initial learning period of seven days during which it learns a new user's activity pattern. The impossible travel detection identifies unusual and

impossible user activity between two locations. The activity should be unusual enough to be considered an indicator of compromise and worthy of an alert. To make this work, the detection logic includes different levels of suppression to address scenarios that can trigger false positive, such as VPN activities. The sensitivity slider allows you to impact the algorithm and define how strict the detection logic is. The higher the sensitivity level, the lower the suppression that is applied as part of the detection logic. In this way, you can adapt the detection according to your coverage needs and your SNR targets.

Kriptik - April 1, 2021, 6:20 am

(Apr 1, 2021 05:44 AM)Shadow_Cloud Wrote: [\[CITE\]](#) You could also send a verification email for logins from a new IP block. So even if someone gets in via either a bf or by a user having the same credentials in a dump somewhere, the 2 factor auth will still protect the account?

With account lockout a spray attack is probably a bigger problem than brute force attacks. If you lock the account until after a password reset after x invalid login attempts, brute force is less likely to be an issue. You could also slow down any brute force attempt drastically by injecting a XSRF token on the login form and only processing logins from valid tokens.

There's a different version of IP 2fa called impossible travel that might be worth considering:

Spoiler ([Click to View](#))

This detection identifies two user activities (is a single or multiple sessions) originating from geographically distant locations within a time period shorter than the time it would have taken the user to travel from the first location to the second, indicating that a different user is using the same credentials. This detection uses a machine learning algorithm that ignores obvious "false positives" contributing to the impossible travel condition, such as VPNs and locations regularly used by other users in the organization. The detection has an initial learning period of seven days during which it learns a new user's activity pattern. The impossible travel detection identifies unusual and impossible user activity between two locations. The activity should be unusual enough to be considered an indicator of compromise and worthy of an alert. To make this work, the detection logic includes different levels of suppression to address scenarios that can trigger false positive, such as VPN activities. The sensitivity slider allows you to impact the algorithm and define how strict the detection logic is. The higher the sensitivity level, the lower the suppression that is applied as part of the detection logic. In this way, you can adapt the detection according to your coverage needs and your SNR targets.

Never heard of Spray Attack before, I just searched Google and apparently it's the same as brute force but attacking multiple accounts at once instead of one account at a time.

I can imagine the issue being that an attacker might just decide to lock other people's account for the fun of it, which I think will be frustrating for other users. In this case I do agree that account lock-out's is not the best solution.

I will for sure give users the possibility to add 2-Factor-Authentication for their account, but I will not force them into doing it. Although if they don't and their account gets hacked, I won't take the blame.

As for preventing brute force attack, reCAPTCHA along with some service like CloudFlare Rate Limiting are I am considering right now. Impossible travel sounds like an interesting solution as well, but I can't find much on the internet, might be because it's a new solution but I will definitely look into it.

Thanks for your input!

Shadow_Cloud - April 1, 2021, 6:26 am

(Apr 1, 2021 06:20 AM)Kriptik Wrote: [\[CITE\]](#) Never heard of Spray Attack before, I just searched Google and apparently it's the same as brute force but attacking multiple accounts at once instead of one account at a time.

I can imagine the issue being that an attacker might just decide to lock other people's account for the fun of it, which I think will be frustrating for other users. In this case I do agree that account lock-out's is not the best solution.

I will for sure give users the possibility to add 2-Factor-Authentication for their account, but I will not force them into doing it. Although if they don't and their account gets hacked, I won't take the blame.

As for preventing brute force attack, reCAPTCHA along with some service like CloudFlare Rate Limiting are I am considering right now. Impossible travel sounds like an interesting solution as well, but I can't find much on the internet, might be because it's a new solution but I will definitely look into it.

Thanks for your input!

Yeah so rather than trying to brute force a single account, imagine an attacker has a combination of 100000 accounts and passwords, he just tried all of them once. To your app it probably looks like legitimate requests to log in.

Yep, but if you only force the 2fa if the login is from a new IP block you should be fine. Even adding a 30 second wait between login requests after x consecutive login failures will make your app way more tedious to brute force and most users shouldn't ever notice. Will the rate limiting work for you if the requests are coming in via randomized proxies? I suspect doing this on an account level will give you better long term results. I think the XSRF token is definitely something you should consider too.

When someone goes to the login page you generate a token.

With the login request, the user has to submit that token too. If you receive a request with an invalid token, you don't process the login request.

After 5 minutes or 1 attempt that token expires.

Kriptik - April 1, 2021, 6:43 am

(Apr 1, 2021 06:26 AM)Shadow_Cloud Wrote: [\[CITE\]](#) Yeah so rather than trying to brute force a single account, imagine an attacker has a combination of 100000 accounts and passwords, he just tried all of them once. To your app it probably looks like legitimate requests to log in.

Yep, but if you only force the 2fa if the login is from a new IP block you should be fine. Even adding a 30 second wait between login requests after x consecutive login failures will make your app way more tedious to brute force and most users shouldn't ever notice. Will the rate limiting work for you if the requests are coming in via randomized proxies? I suspect doing this on an account level will give you better long term results. I think the XSRF token is definitely something you should consider too.

When someone goes to the login page you generate a token.

With the login request, the user has to submit that token too. If you receive a request with an invalid token, you don't process the login request.

After 5 minutes or 1 attempt that token expires.

At first I didn't know what XSRF was, so I went to look it up and apparently XSRF and CSRF are the same. I already have this implemented for all my forms, I am currently developing my app on Flask and I am using the Flask WTFForms which automatically enables CSRF protection. So I

currently have this one covered.

(Apr 1, 2021 06:26 AM)Shadow_Cloud Wrote: [\[CITE\]](#) Yep, but if you only force the 2fa if the login is from a new IP block you should be fine. Even adding a 30 second wait between login requests after x consecutive login failures will make your app way more tedious to brute force and most users shouldn't ever notice.

Yes this is one of the things I'm trying to do, but I'm worried about attackers using different proxies with their tools.

Shadow_Cloud - April 1, 2021, 6:50 am

(Apr 1, 2021 06:43 AM)Kriptik Wrote: [\[CITE\]](#) At first I didn't know what XSRF was, so I went to look it up and apparently XSRF and CSRF are the same. I already have this implemented for all my forms, I am currently developing my app on Flask and I am using the Flask WTFForms which automatically enables CSRF protection. So I currently have this one covered.

Yes this is one of the things I'm trying to do, but I'm worried about attackers using different proxies with their tools.

As long as you aren't processing requests with invalid XSRF tokens, brute force will be a lot harder, because the attacker would need to fetch a token for each request.

I've implemented the auth features for some very large systems, including banks, if you need help in this space feel free to give me a shout. What you want to do is introduce more and more gradual delays as login failure count increases. I know ideally you want to just lock the account, but that's not always commercially feasible. DDOS attacks at layer 7 follows very much down this principle where legitimate requests and the attacker's requests look the same, so its extremely hard to block the attack without blocking legitimate users too.

Kriptik - April 1, 2021, 7:10 am

(Apr 1, 2021 06:50 AM)Shadow_Cloud Wrote: [\[CITE\]](#) As long as you aren't processing requests with invalid XSRF tokens, brute force will be a lot harder, because the attacker would need to fetch a token for each request.

I've implemented the auth features for some very large systems, including banks, if you need help in this space feel free to give me a shout. What you want to do is introduce more and more gradual delays as login failure count increases. I know ideally you want to just lock the account, but that's not always commercially feasible. DDOS attacks at layer 7 follows very much down this principle where legitimate requests and the attacker's requests look the same, so its extremely hard to block the attack without blocking legitimate users too.

I will sure contact you if I struggle with this, however thanks so much for providing all of this valuable information as this will surely help me with my project.