You are participating in a **private** program for **Linode**. Please do **not** publicly discuss the program until the program goes public.

3

SSRF in Managed Service Functionality

State	○ Resolved (Closed)
Disclosed	**January 8, 2019 10:54pm +0530**
Reported To	Linode
Asset	https://manager.linode.com (Domain)
Weakness	Server-Side Request Forgery (SSRF)
Bounty	$350
Severity	Medium (5.9)
Participants	
Visibility	Disclosed (Full)

Collapse

rhynorater submitted a report to **Linode**. Aug 25th (5 months ago)

Hello,

Thanks for sponoring H1-702! I'm wearing some Linode gear as I'm writing this :)

Your managed service monitor functionality has a feature where you can set a URL or a hostname and port to be checked for service integrity. While it seems 127.0.0.1 is blocked to prevent SSRF, any domain that resolves to 127.0.0.1 allows for a bypass and allows an attacker to request internally.

Thanks to the "Response body match" functionality, an attacker can receive the response to a binary answer. For example, I was able to make a request to 127.0.0.1.xip.io:22 and check the response body for OpenSSH and the monitor service gave the "OK" response. To replicate this, please see the following steps:

1. Start up a linode
2. Visit `https://manager.linode.com/managed/monitoring`
3. Add a service with `Monitor Type` set to TCP and `TCP` set to `127.0.0.1.xip.io:22` and `Response body match` set to `OpenSSH`
4. Submit the request, and wait until the service moves from Pending to OK.

I've also been able to determine that there is a service open on port 25. I could possibly use this to send email. Will work on a working POC for impact. Standby.

Impact

Internal port scanning, interfacing with internal service.

joystick HackerOne staff posted a comment. Aug 25th (5 months ago)

Hi,

Thank you for your submission. We are investigating your report, and will let you know if we have any questions. We appreciate your assistance and cooperation.

Regards.
@joystick

rhynorater posted a comment. Aug 25th (5 months ago)

We were also able to determine that there is a redis server on 6379 and a smtp server on 25.

We are working to exploit this now.

rhynorater posted a comment. Aug 26th (5 months ago)

I was successfully able to identify the internal SSH service as `OpenSSH_6.0`

rhynorater posted a comment. Aug 26th (5 months ago)

Just to elaborate on this further - there appears to be an internal redis server. However, I cannot exploit this due to the fact that I cannot get the request (via redirects) to send any commands with a newline. If I could simply get ONE newline, then I could execute the "SLAVEOF" command on this redis server allowing me to assign my machine as the master and control it. There is risk of compromise here, but I cannot find a valid exploit as all I have access to is GET requests.

I was also unable to send email via SMTP for the same reason, though I have been told this is possible via HTTPS somehow.

While this service does support following redirects, it doesn't allow any other protocols besides http and https to be used :/

thefrog changed the status to ○ **Triaged**. Aug 26th (5 months ago)

Hi @rhynorater,

Thank you for your submission. I was able to validate your report, and have submitted it to the appropriate team for a thorough evaluation. The report will be reviewed to assess the full impact, after which a final ruling will be made. Please note that the status and severity of this report are subject to change.

Best regards,
@thefrog

Security Analyst
HackerOne

rhynorater posted a comment. Aug 26th (5 months ago)

Thanks for the triage @thefrog!

linode_ctarquini updated the severity from High to Medium (5.9). Aug 29th (5 months ago)

Linode rewarded rhynorater with a **$300** bounty and a **$50** bonus. Aug 29th (5 months ago)

linode_ctarquini posted a comment. Aug 29th (5 months ago)

Hey @rhynorater,

Thanks again for your report. We have a patch in the works to prevent the managed monitors from accessing internal networks. We've marked this as a medium for now as it seems that the scope is limited the ability to inject commands into the target.

Let us know if you find anything else!

rhynorater posted a comment. Aug 29th (5 months ago)

Thanks for the bounty and the bonus! @linode_ctarquini

rhynorater posted a comment. Aug 29th (5 months ago)

Please let me know when a patch is in place. I've got some bypasses up my sleeve :)

linode_ctarquini posted a comment. Sep 5th (5 months ago)

Hey @rhynorater,

Can you provide the username associated with your H1 Linode account? I'd like to set you up with a monitor that will have the patch applied so you can validate it looks good

rhynorater posted a comment. Sep 5th (5 months ago)

@linode_ctarquini Sure! It's hackerone16

linode_ctarquini posted a comment. Sep 5th (5 months ago)

Hey @rhynorater,

I've moved your monitor for local openssh to a patched environment. If you enable it now, you'll see it now fails.

Our mitigation runs these checks in a separate network namespace so that the host `lo` interface (where redis and ssh are bound) is not available to the process running the checks. I believe this should be enough to thwart bypasses dependent on redirection or DNS rebind attacks since the process running checks simply can't route traffic outside it's own isolated version of `lo`.

linode_ctarquini closed the report and changed the status to ○ **Resolved**. Sep 7th (5 months ago)

Hey @rhynorater,

We've deployed the patch to our entire fleet now and you should no longer be able to reach internal networks via the monitors.

I'm going to close this out for now but if you bypass it, please let me know and we'll hook you up with another bounty :).

rhynorater posted a comment. Sep 7th (5 months ago)

Hey @linode_ctarquini! Just to let you know, I did go ahead and verify this fix - I was unable to bypass it. Sorry that I didn't get back to you. Thanks again!

linode_bdorsey requested to disclose this report. Jan 8th (22 days ago)

Hi @rhynorater, the Linode Security Team is interested in requesting disclosure of this high-quality report to the rest of the members of our private bug bounty program.

rhynorater agreed to disclose this report. Jan 8th (22 days ago)

This report has been disclosed. Jan 8th (22 days ago)