A10-Unvalidated Redirects and Forwards

Exploitability: AVERAGE | Prevalence: COMMON | Detectability: EASY | Technical Impact: MODERATE

Description

Web applications frequently redirect and forward users to other pages and websites, and use untrusted data to determine the destination pages. Without proper validation, attackers can redirect victims to phishing or malware sites, or use forwards to access unauthorized pages.

Attack Mechanics

An attacker can use unvalidated redirected links as a medium to redirect user to malicious contents and tricks victims into clicking it. Attacker can exploit it to bypass security checks and make it believe trustworthy.

For example, the "Learning Resources" link (`/learn?url=...`) in the application redirects to another website without validating the url.

Here is code from `routes/index.js` ,

```
// Handle redirect for learning resources link
app.get("/learn", function (req, res, next) {
    return res.redirect(req.query.url);
});
```

An attacker can change the `url` query parameter to point to malicious website and share it. Victims are more likely to click on it, as the initial part of the link (before query parameters) points to a trusted site.

How Do I Prevent It?

Safe use of redirects and forwards can be done in a number of ways:

1. Simply avoid using redirects and forwards.
2. If used, don't involve user parameters in calculating the destination. This can usually be done.
3. If destination parameters can't be avoided, ensure that the supplied value is valid, and authorized for the user.
   It is recommended that any such destination parameters be a mapping value, rather than the actual URL or portion of the URL, and that server side code translate this mapping to the target URL.