Securing Development that's Gone Remote

snyk

1. Empower developers with documented guidelines

Allow each developer to make progress on their own, and keep communication async.

Invest in documenting your practices and expectations, making it easier for developers to make the right decisions consistently.

2. Instead of breaking the build, fail pull requests

"Breaking the build" due to a security violation is a very popular CI/CD security measure—but it is also very disruptive.

Fail pull requests instead. Pull requests have several advantages:

- They allow you to test only the new code changes, which should be within the developer's control to fix.
- They're more local to the branch where code is modified, maintaining individual developer autonomy.
- You can choose whether a given failure blocks a merge or is just informational, again allowing developers to make the call and proceed.

3. Invest in security visibility

Visibility comes in many shapes, but here are a few suggestions:

- Instrument builds to capture the dependencies packaged into your app.
- Post vulnerabilities discovered in the build (that weren't severe enough to break it).
- Create leaderboards showing how well are different teams handling security issues. Security achievements.

Some tools to consider:

Snyk , Okta, Duo, HackerOne and Bugcrowd

4. Improve individual skills

Working remotely can be an exercise in self reliance, but it can also create additional time in people's days. Use that as an opportunity to level up your teams, both on the dev and security side:

- Encourage attendance at virtual conferences that being offered, most for free
- Foster sharing of free online resources such as instructional videos, blogs, etc.
- Set objectives for talent growth during this time of increased remote working.

5. Celebrate security wins

Make sure you give developers a virtual pat on the shoulder when they advance the security cause.

- A kind word on Slack or group email. This is simple, but important. It can be one-off or some sort of "Security champion of the month"—as long as you keep doing it. You can even recognize their work publicly on your company's social account if it merits it.
- Special swag, ranging from stickers to t-shirts to hoodies, reserved for those who've earned it. There are lots of examples of this on The Secure Developer podcast.
- Actual monetary value gift, like a SPA day or a trip and ticket to DefCon (if conferences still exist after the dust settles).

6. Increase security and development engagement

Connect resources from security and development in daily working practices. Match up your development and security resources:

- Schedule a regular cadence of meetings between paired peers across development and security
- Drive security participation in daily stand-ups and sprint meetings

7. Focus on security hygiene

Our new reality requires focus, ensuring what matters most gets done. In security, that means prioritizing the basics before the esoteric attacks. Focus your efforts first on:

- Identifying and addressing vulnerable components
- Detecting configuration mistakes
- Protecting secrets sufficiently

8. Invest in Multi-Factor Authentication

Embrace this as an opportunity to begin investing in 2 factor authentication infrastructure.

- If you're not already, implement MFA to protect remote access endpoints
- Extend MFA to critical systems and applications
- Begin working toward a zero trust model in identity management

9. Improve SSH Security

As more of those machines go remote, the risk of attack on SSH sessions used for code repository interactions goes up. Take the following steps to secure those sessions:

- Enable mutual key-based authentication
- Enable or reduce session timeouts
- Enable stronger identity based authentication

10. Bug Bounties

There are many people out of work who are turning to gig jobs as a way to make ends meet. This about a bug bounty program to:

- Add an additional layer of security assessment capability
- Provide clear communication expectations for vulnerability reports
- Help support researchers in crowdsourced programs