Experiences Deploying Multi-Vantage-Point Domain Validation at Let’s Encrypt

Henry Birge-Lee
Liang Wang, Daniel McCarney, Roland Shoemaker, Jennifer Rexford, Prateek Mittal
Routing Attacks on Certificate Authorities

- Trust on the Internet relies on valid digital certificates.
- Domain validation is used to verify identity during certificate issuance.
- Adversaries can manipulate Internet routing to direct validation challenge to wrong party.
- Adversaries can obtain malicious certificates and man in the middle TLS connections.

1. Certificate Request
2. DV Challenge
3. Launch localized BGP attack
4. Verify challenge
5. Certificate

Legitimate Server

Certificate Authority

Adversary
Multiple vantage point domain validation

1. Certificate Request
2. DV Challenge
3. Launch localized BGP attack
4. Verify challenge
5. Error: Challenge not completed at remote vantage point

- Raises the bar for an adversary by forcing the adversary to attack multiple vantage points with a global (not localized) attack.
Multiple vantage point domain validation

We Designed and Deployed Multiple Vantage Point Domain Validation at Let’s Encrypt (the world’s largest publicly trusted CA)
Design Questions

- Where should vantage points be located?
- How should results from vantage points be considered?
- How should the multiple vantage points be integrated into the Let’s Encrypt code base?
Where should vantage points be located?

- Billing and management are simplified if all vantage points are in the same cloud provider.
- But does a single cloud provider still provide sufficient security?
  - Yes! If vantage points are spread out geographically they will use local peering connections to route to the victim and will have sufficient diversity.
- Vantage points are in the US West Coast, East Coast and Europe.
How should results be considered?

- Requiring validation from all vantage points could cause a single outage to take down the CA
- The CA/Browser Forum (that govern CAs) has auditing requirements that are not enforced at remote vantage points
How should the multiple vantage points be integrated into the Let’s Encrypt code base?

- Let’s Encrypt has a modular code base connected by RPC calls (which are authenticated and secure over the public Internet)
- Much of the code from the original validation authority (VA) can be reused for multiVA
- MultiVA is a extension to the original VA module (leaving other parts of the code unaffected)
  - Only ~200 lines of code needed for core multiVA logic
- Open-source code:
  - [https://github.com/letsencrypt/boulder/blob/main(va)/va.go](https://github.com/letsencrypt/boulder/blob/main(va)/va.go)
Preventing disruptions with a phased deployment

- **Phase 1**: Monitoring only: multiVA results are logged but do not influence validation
  - Used to test mutiVA code at scale and log any potential errors and estimate costs (~$100 per month per VA)
- **Phase 2**: Enforcing with exception list: multiVA is enforced except on certain accounts
- **Phase 3**: Full deployment
Preventing disruptions with a phased deployment

Since enforcement, over half a billion certificates have been signed with multiVA

- Phase 2: Enforcing with exception list: multiVA is enforced except on certain accounts
- Phase 3: Full deployment
Benign Failures

Benign Failures:
Certificates that blocked because of multiVA where no attack was present
Understanding Benign Failures

- Causes of benign failures:
  - Domain registration and certificate request often happen around the same time
  - DNS may not have fully propagated causing failures at remote vantage points

- Mitigating benign failures:
  - Use a quorum policy (e.g., allow one vantage point to fail)
  - Customers can try again: ~50% of failed certificates are retried and ultimately signed
Benign Failure rate is manageable at Internet scale

- Use a quorum policy (e.g., allow one vantage point to fail)
- Customers can try again: ~50% of failed certificates are retried and ultimately signed
Evaluating Security via Real-World Attacks

- Tested with real-world BGP attacks on domains we controlled and certificates requested from Let’s Encrypt
- Ethical Considerations:
  - Performed using the PEERING testbed
  - Had authorization from PEERING to announce IP prefix at all PEERING nodes
  - IP prefixes-domains were only used for this experiment (i.e., no real users or services)
Real world attacks with PEERING testbed

Before Attack:

Internet

Primary DC

Ohio

Frankfurt

Oregon

1.2.3.0/24

Wisc01 Victim

Neu01 Adversary
Real world attacks with PEERING testbed

After Attack:

1.2.3.0/24

Wisc01 Victim

Oregon

Frankfurt

Ohio

Primary DC

Certificate Request

1.2.3.0/24

Neu01 Adversary
Real world attacks with PEERING testbed

Certificate Issuance Block Because of Multiple Vantage Point Validation

After Attack:

1.2.3.0/24 Certificate Request

Certificate Issuance Blocked Because of Multiple Vantage Point Validation

1.2.3.0/24

Frankfurt

Ohio

Wisc01 Victim

Neu01 Adversary
Considering more adversaries shows security improvement

- Considered all six other active PEERING nodes as adversaries attacking wisc01
- Without multiVA: five out of six attacks succeeded
- With multiVA: only one out of the six attacks succeeded
Evaluating security via simulations

● Measured “resilience”:
  The fraction of ASes on the Internet that are topologically incapable of launching a BGP attack to get a certificate for a domain.

● MultiVA increased median domain resilience from .62 to .95
● Even greater improvement on 10th percentile domain from .10 to .51
● More vantage points further improve resilience
Conclusion

- MultiVA design is feasible at Internet scale (i.e., deployed at Let’s Encrypt with more than half a billion certs issued)
- Next steps: pushing for more CA adoption and ultimately full industry adoption
- Code is open source and available at:
Conclusion

- MultiVA design is feasible at Internet scale (i.e., deployed at Let’s Encrypt)
- Next steps: pushing for more CA adoption and ultimately full industry adoption
- Code is open source and available at:

Questions?

Henry Birge-Lee
birgelee@princeton.edu