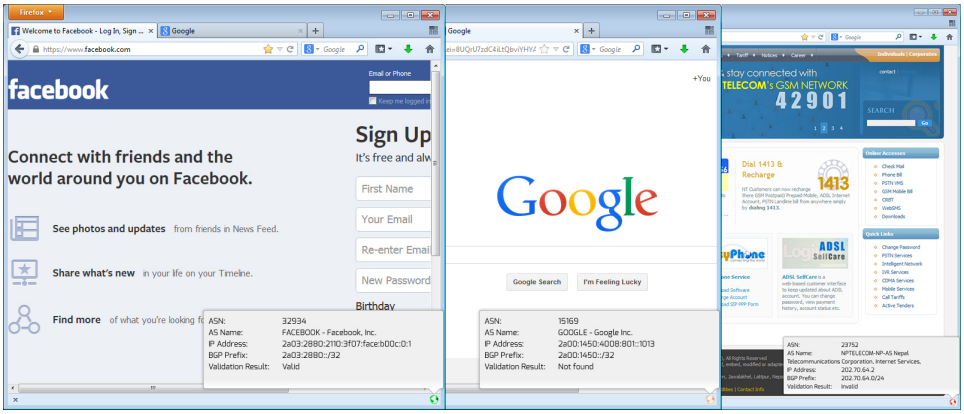

Figure 1: RPKI validation in Mozilla Firefox for the requested websites with different validation outcome

the BGP data depend on the location of the client. However, in contrast to the name to address mapping in the DNS, there is no standard mechanism to request the prefix/AS pair for an IP address with respect to the customer's ISP routing table. An accurate mapping might be a new service provided by ISPs in the future.

There are two options to implement origin validation in web browsers: (a) the browser extension implements the full router part (i.e., receives valid ROAs from cache server and performs origin validation of BGP data), (b) the extension resolves only the IP address of the web domain and a remote back-end performs the origin validation. We decide for the latter as this allows for easy applicability in most browser platforms, which usually provide add-on concepts based on JavaScript. Back-end and front-end communicate via HTTP as this is native in browsers and does not conflict with most firewall settings.

Implementation—Back-end Currently, the back-end uses the Team Cymru community service to resolve the IP prefix of the web server IP address and the corresponding origin AS. We admit that the result does not necessarily comply with the BGP entry of the client's upstream but the vantage points of Team Cymru provide a good coverage. Furthermore, our architecture is flexible enough to consider multiple BGP sources as well as future mappig services.

To fetch ROAs and to validate the BGP information, we deploy the RTRlib [5], an open source implementation of the RPKI/RTR router part. This C library is very efficient with respect to memory and processing resources. Per default, the implementation establishes RTR sessions to two cache servers for fallback reasons. However, end users can configure their own end point for a cache server in the browser extension, and multiple instances of the RTRlib will be started.

Implementation—Front-end The browser extension is implemented as dynamic add-on for Mozilla Firefox and Chrome. Other browsers can be easily supported as the browser extension only needs to support the REST interface to the back-end. The source code is available on Github[1]. The extension visualizes three states: green (the web server prefix is valid in the BGP), orange (the prefix was not found in the RPKI), and red (the prefix is invalid, the website might be suspicious), see Fig. 1. Note that if an attacker blocks plugin traffic, none of the three states apply, indicating that the communication to the trust anchor is broken. Advanced users can request information about the autonomous system/the IP prefix and configure the host address and port of the RPKI cache server, which will be used by the back-end.

[1]https://github.com/rtrlib

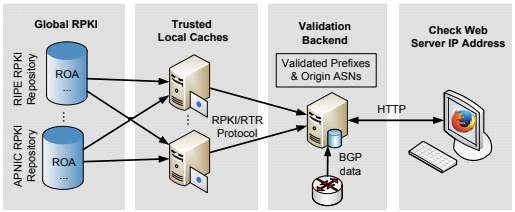

Figure 2: System architecture

Future Challenges Our current solution checks the BGP data for the web server infrastructure of the landing page. Many web pages include embedded content linked via different domains. Analyzing the HTML content fast and combine the different results to a complete picture for the whole web page will be part of our future work.

Acknowledgments We would like to thank Tas Sóti and Sebastian Meiling for helping on the implementation. This work is supported by the German BMBF within the project Peeroskop (http://peeroskop.realmv6.org).

4. REFERENCES

[1] Bush, R., and Austein, R. The Resource Public Key Infrastructure (RPKI) to Router Protocol. RFC 6810, IETF.

[2] Bush, R., Austein, R., Patel, K., Gredler, H., and Waehlisch, M. Resource Public Key Infrastructure (RPKI) Router Implementation Report. RFC 7128, IETF, 2014.

[3] Iamartino, D., Pelsser, C., and Bush, R. Measuring BGP route origin registration validation . In *Proc. of PAM* (Berlin, 2015), LNCS, Springer, pp. 28–40.

[4] Mohapatra, P., Scudder, J., Ward, D., Bush, R., and Austein, R. BGP Prefix Origin Validation. RFC 6811, 2013.

[5] Wählisch, M., Holler, F., Schmidt, T. C., and Schiller, J. H. RTRlib: An Open-Source Library in C for RPKI-based Prefix Origin Validation. In *Proc. of USENIX Security Workshop CSET'13* (Berkeley, 2013), USENIX Assoc.

[6] Wählisch, M., Maennel, O., and Schmidt, T. C. Towards Detecting BGP Route Hijacking using the RPKI. *ACM Computer Communication Review 42*, 4 (2012), 103–104.

[7] Wählisch, M., Schmidt, R., Schmidt, T. C., Maennel, O., and Uhlig, S. When BGP Security Meets Content Deployment: Measuring and Analysing RPKI-Protection of Websites. Technical Report arXiv:1408.0391, Sep. 2014.