Jianjun Chen

Contact Information

Address:	1947 Center Street, Suite 600, Berkeley, CA, 94704
email:	jianjun@icsi.berkeley.edu
Phone:	(+1) 510-631-6085

Research Interest

Network security, protocol security, web security, vulnerability discovery

Education

International Computer Science Institute — Berkeley, CA
Postdoc in Networking and Security Group — Aug 2018 – Present
Supervisor: Prof. Vern Paxson

Tsinghua University — Beijing, China
Ph.D. in Computer Science and Technology — Sep 2013 – Jul 2018
Supervisor: Prof. Haixin Duan

Wuhan University — Wuhan, China
B.E. in Computer Science and Technology — Sep 2009 – Jul 2013

Publications

- *Jianjun Chen*, V. Paxson, J. Jiang. Composition Kills: A Case Study of Email Sender Authentication, In Proceedings of the 29th USENIX Conference on Security Symposium (USENIX Security'20), August 2020. (**Distinguished Paper Award**). (Also presented at Black Hat USA 2020).

- *Jianjun Chen*, J. Jiang, H. Duan, T. Wan, S. Chen, V. Paxson, M. Yang. We Still Don't Have Secure Cross-Domain Requests: an Empirical Study of CORS, In Proceedings of the 27th USENIX Conference on Security Symposium (USENIX Security'18), August 2018.

- *Jianjun Chen*, J. Jiang, X. Zheng, H. Duan, J. Liang, K. Li, T. Wan, and V. Paxson, Forwarding-Loop Attacks in Content Delivery Networks, Network and Distributed System Symposium (NDSS'16), February 2016. (**Distinguished Paper Award**).

- *Jianjun Chen*, J. Jiang, H. Duan, N. Weaver, T. Wan, and V. Paxson. Host of Troubles: Multiple Host Ambiguities in HTTP Implementations, In Proceedings of the 23rd ACM SIGSAC Conference on Computer and Communications Security (CCS'16), October 2016.

- R. Guo, W. Li, B. Liu, S. Hao, J. Zhang, H. Duan, K. Shen, *Jianjun Chen*, and Y. Liu. CDN Judo: Breaking the CDN DoS Protection with Itself. Network and Distributed System Symposium (NDSS'20), February 2020.

- Guo, R., *Jianjun Chen*, Liu, B., Zhang, J., Zhang, C., Duan, H., and Jia, Y. Abusing CDNs for Fun and Profit: Security Issues in CDNs' Origin Validation. In the Proceedings of the IEEE 37th Symposium on Reliable Distributed Systems (SRDS'18), October 2018.

- Chen, F., Duan, H., Zheng, X., Jiang, J., and *Jianjun Chen*. Path Leaks of HTTPS Side-Channel by Cookie Injection. In International Workshop on Constructive Side-Channel Analysis and Secure Design (pp. 189-203). April 2018.

- X. Liao, K. Yuan, X. Wang, Z. Pei, H. Yang, *Jianjun Chen*, H. Duan, K. Du, E. Alowaisheq, S. Alrwais, L. Xing, and R. Beyah. Seeking Nonsense, Looking for Trouble: Efficient Promotional-Infection Detection through Semantic Inconsistency Search. In the Proceedings of the 37th IEEE Symposium on Security & Privacy (Oakland'16), May 2016.

Research/Internship Experience

Nov 2017 – Feb 2018 **Nanyang Technological University, Singapore**
Research Intern; Advisor: Prof. Yang Liu

Dec 2016 – Jun 2017 **Baidu, Beijing**
Research Intern; Responsible for anti-phishing prototype system development

Sep 2015 – Mar 2016 **International Computer Science Institute, CA**
Research Intern; Advisor: Prof. Vern Paxson

Sep 2013 – Jul 2015 **Tsinghua Univeristy, Beijing**
Teaching Assistant; Course: Network and System Security

Research Projects

Aug 2018 – Present **Security Analysis of Email Systems** ICSI
Investigated how email authentication mechanisms like SPF/DKIM/DMARC were implemented in practice, and discovered a number of email spoofing bugs affecting popular email providers (e.g., Gmail, iCloud) and clients (e.g., Thunderbird, Outlook). This work is covered by Wired [10], CSO online [4], and Dark reading [8].

Apr 2017 – May 2018 **Security Analysis of CORS Protocol** Tsinghua
Performed a security analysis of the CORS (Cross Origin Resource Sharing) protocol, and discovered multiple security issues, which have led to both web standard [15] and major browsers (e.g., Chrome [5], Firefox [7], Webkit [2, 3]) modification.

Aug 2015 – Jun 2016 **Security Analysis of Middle-boxes in HTTP Systems** ICSI
Discovered a class of new HTTP smuggling attacks affecting a wide range of HTTP systems, which can cause HTTP cache poisoning and firewall bypass. This work has led to patches and email acknowledgments by Squid [13, 14], Akamai, Alibaba CDN, Tencent CDN, ESET, Palo Alto Networks firewall, and Huawei firewall.

Sep 2014 – Oct 2015 **Security Analysis of Content Delivery Networks (CDN)** Tsinghua
Identified an architecture weakness in Content Delivery Networks (CDN), which allowed attackers to launch Denial-of-Service attacks against CDN itself. The work has led to security advisories or blogs by CDN vendors (e.g., Akamai [1], Cloudflare [6, 12], Fastly [11]) and a new IETF RFC (RFC 8586 [9]).

Selected vulnerabilities

CVE-2016-4553	Squid team evaluated it as a highest level (blocker) security vulnerability. An attacker may remotely poison the cache of *any* HTTP website with arbitrary content.
CVE-2016-4554	A critical security vulnerability in Squid, which was introduced to version 1.0 in 1996
VU#938151	Potential Denial of Service attacks affecting 16 CDN vendors
CVE-2018-4295	A web attacker may be able to attack macOS AFP servers through browser JavaScript.
CVE-2018-8014	CORS misuse in Apache Tomcat. Other similar issues reported by me, PHP Yii2 (CVE-2018-20745), Go-CORS (CVE-2018-20744).

Awards and Scholarships

2020	**Distinguished Paper Award**, USENIX Security 2020
2019	**ACM China SIGSAC Doctoral Dissertation Award**, ACM China
2017	**Network Security Scholarship**, China Internet Development Foundation
2016	**Distinguished Paper Award**, Network and Distributed System Symposium (NDSS)
2012	**National Scholarship**, Ministry of Education, China
2011	**National Endeavor Scholarship**, Ministry of Education, China

Academic Activities

- Reviewer, ACM Conference on Computer and Communications Security (CCS), 2019
- Reviewer, European Symposium on Research in Computer Security (ESORICS), 2019
- Reviewer, IEEE/ACM Transactions on Networking (ToN), 2018

References

[1] Akamai. Akamai Response to Forwarding-loop Issue. https://blogs.akamai.com/2016/03/akamai-response-to-forwarding-loop-issue.html, 2016. [accessed Apr-2019].

[2] Apple. CVE-2018-4295. https://support.apple.com/en-us/HT209193, 2018. [accessed Apr-2019].

[3] bfulgham@apple.com. Add port 548 (afpovertcp) to port blacklist. https://git.webkit.org/?p=WebKit.git;a=commit;h=02b6d273eff5652fb058bd3e8d276df9c6ca0202, 2018. [accessed Apr-2019].

[4] Brumfield, C. 18 (new) ways attackers can compromise email. https://www.csoonline.com/article/3570421/18-new-ways-attackers-can-compromise-email.html, Aug 2020.

[5] Chen, J. Issue 824130: Several CORS security issues in browsers and specs. https://bugs.chromium.org/p/chromium/issues/detail?id=824130, 2018. [accessed Apr-2019].

[6] Davidson, A. Preventing Request Loops Using CDN-Loop. https://blog.cloudflare.com/preventing-request-loops-using-cdn-loop/, 2019. [accessed Aug-2020].

[7] Kingston, J. Implement stricter CORS checking for headers. https://hg.mozilla.org/mozilla-central/rev/a46028ac9dbb, 2018. [accessed Apr-2019].

[8] Lemos, R. Email Security Features Fail to Prevent Phishable 'From' Addresses. https://www.darkreading.com/vulnerabilities---threats/email-security-features-fail-to-prevent-phishable-from-addresses/d/d-id/1338448?#msgs, Aug 2020.

[9] Ludin, S., Nottingham, M., and Sullivan, N. Loop Detection in Content Delivery Networks (CDNs). RFC 8586, IETF, 2019.

[10] NEWMAN, L. H. Decades-Old Email Flaws Could Let Attackers Mask Their Identities. https://www.wired.com/story/decades-old-email-flaws-could-let-attackers-mask-identities/, Aug 2020.

[11] Perez, M. What is a CDN and why you should use one. https://www.fastly.com/blog/why-you-should-use-content-delivery-network, 2016. [accessed Aug-2020].

[12] Sullivan, N. Preventing Malicious Request Loops. https://blog.cloudflare.com/preventing-malicious-request-loops/, 2016. [accessed Apr-2019].

[13] Team, S. Squid Proxy Cache Security Update Advisory SQUID-2016:7. http://www.squid-cache.org/Advisories/SQUID-2016_7.txt, May 2016.

[14] Team, S. Squid Proxy Cache Security Update Advisory SQUID-2016:8. http://www.squid-cache.org/Advisories/SQUID-2016_8.txt, May 2016.

[15] van Kesteren, A. Strengthen requirements on CORS-safelisted request-headers, https://github.com/whatwg/fetch/pull/736. Fetch standard, Web Hypertext Application Technology Working Group (WHATWG), 2018.