Ensemble Approach to Failure-Resistant Password-Based Key Derivation Functions

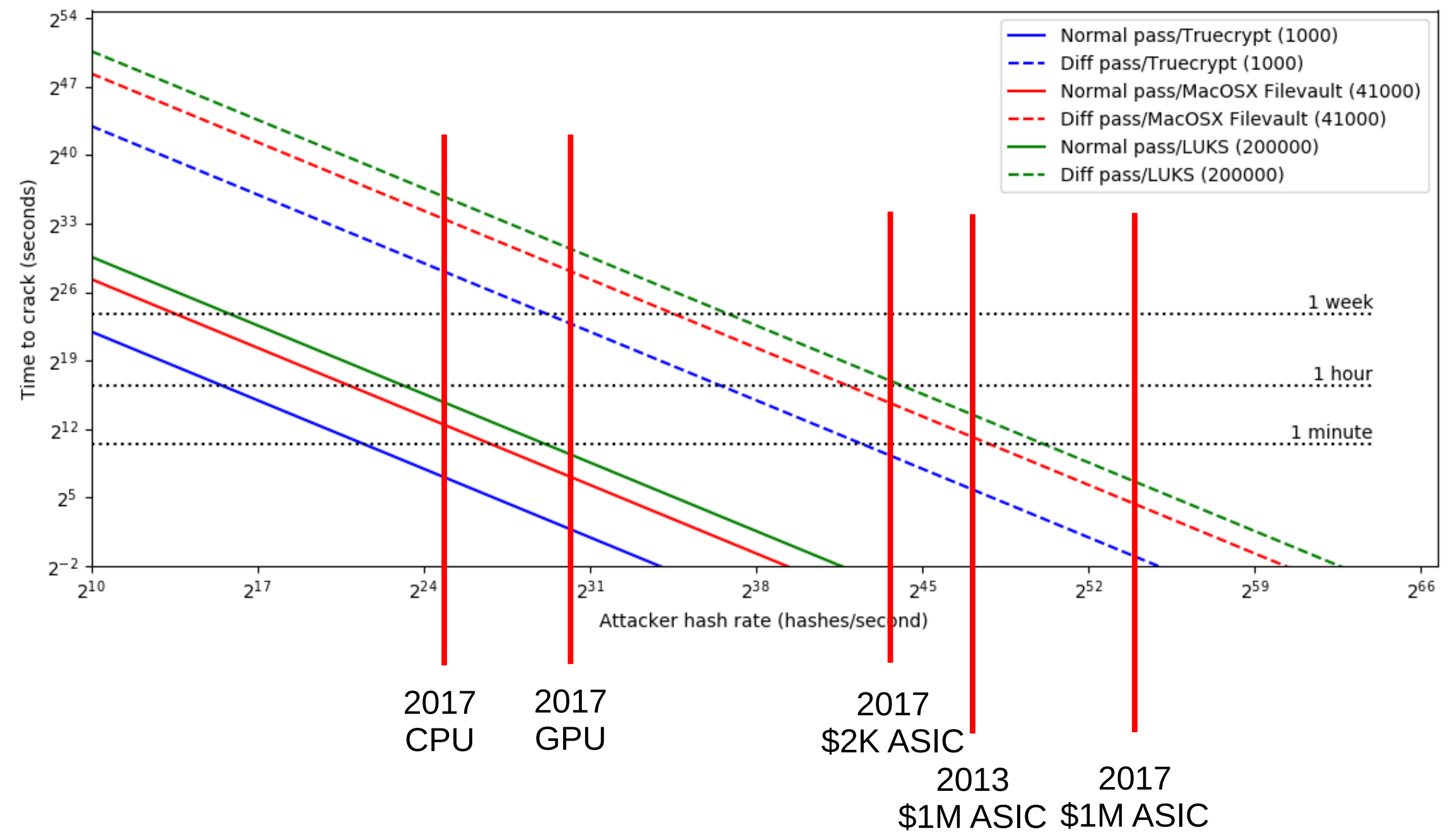
Jason Hennessey, Sarah Scheffler, Mayank Varia

{henn, sscheff, varia}@bu.edu

BU Department of Computer Science

bu.edu/cs

Motivation



- PBKDF2[13] (most widespread PBKDF) relies on simple, repeated hash invocations to increase password key derivation time for attackers
- Bitcoin provided a financial incentive to create high throughput, efficient hashing ASICs
- Passwords can now be guessed 10^6 to 10^{10} times faster using ASICs than CPUs of similar price
- State-of-the-art PBKDFs (e.g. scrypt[8], argon2[4]) improve by utilizing memory, but are still vulnerable to ASIC attacks [1]

Goal

Minimize efficiency gains of specialized hardware vs. honest user's device for key derivation

Properties

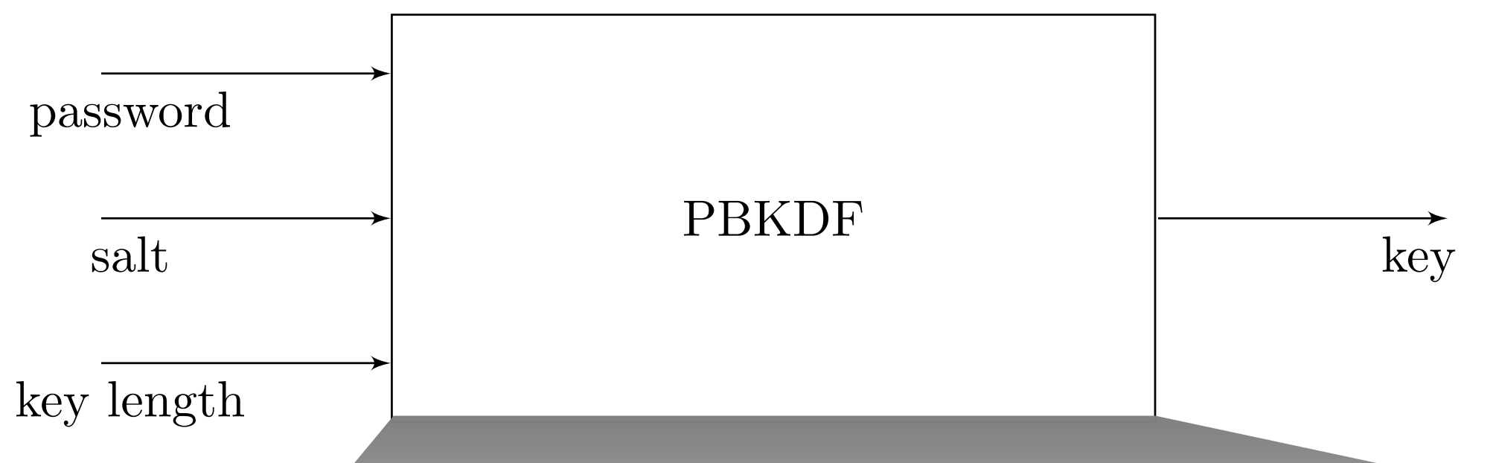
- **Resource consumption model** - plugins consume user-specified resources (e.g. memory, CPU, disk)
- **Failure resistance** - Hash construct guarantees security as good as strongest hash; failures in resource-consuming plugins limited to a single round
- **Optimization for specific platform** - Plugin and sponge construction designed for anti-pipelining and anti-parallelism

Acknowledgements

We gratefully acknowledge Ethan Heilman's contributions both to the initial concept and refinements, as well as support received from: NSF Grant No. 1414119, Clare Boothe Luce Graduate Research Fellowship, MassTech Collaborative Research Matching Grant Program and commercial partners of the Massachusetts Open Cloud.

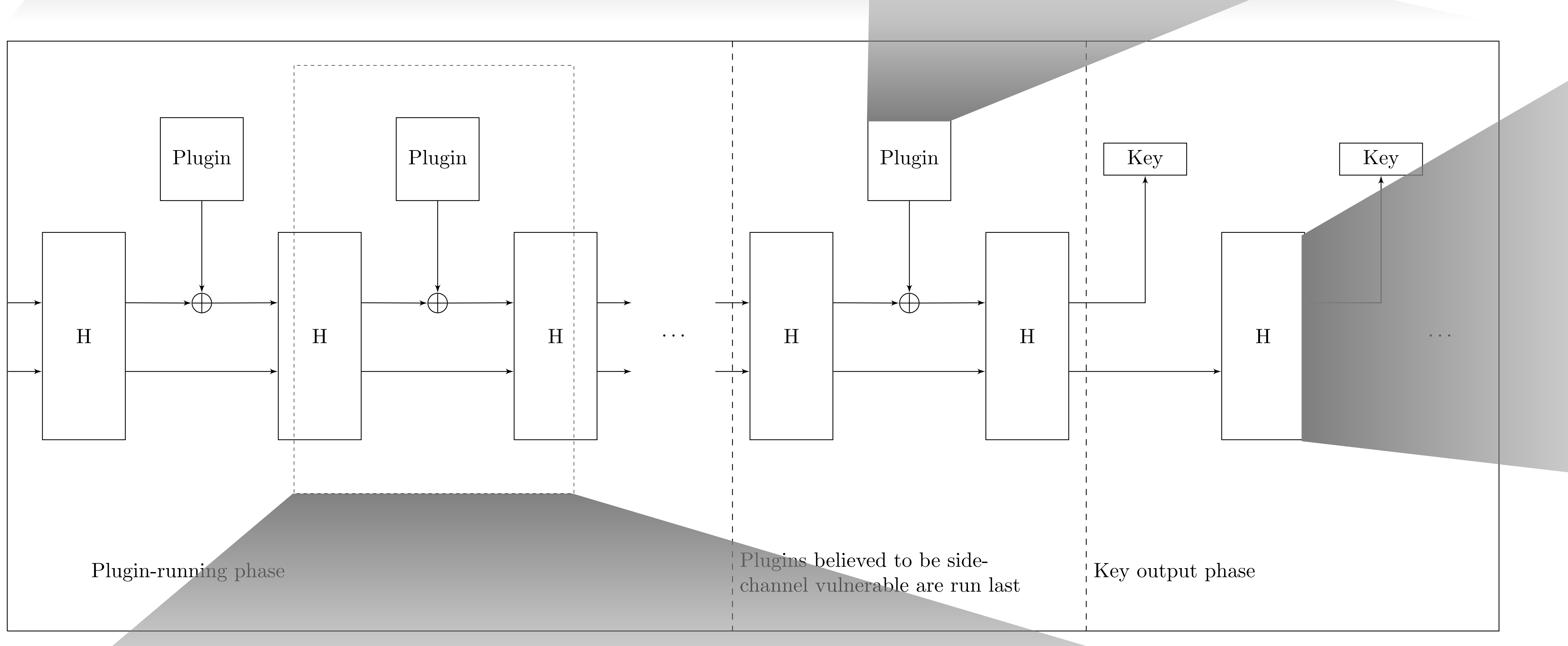
Construction

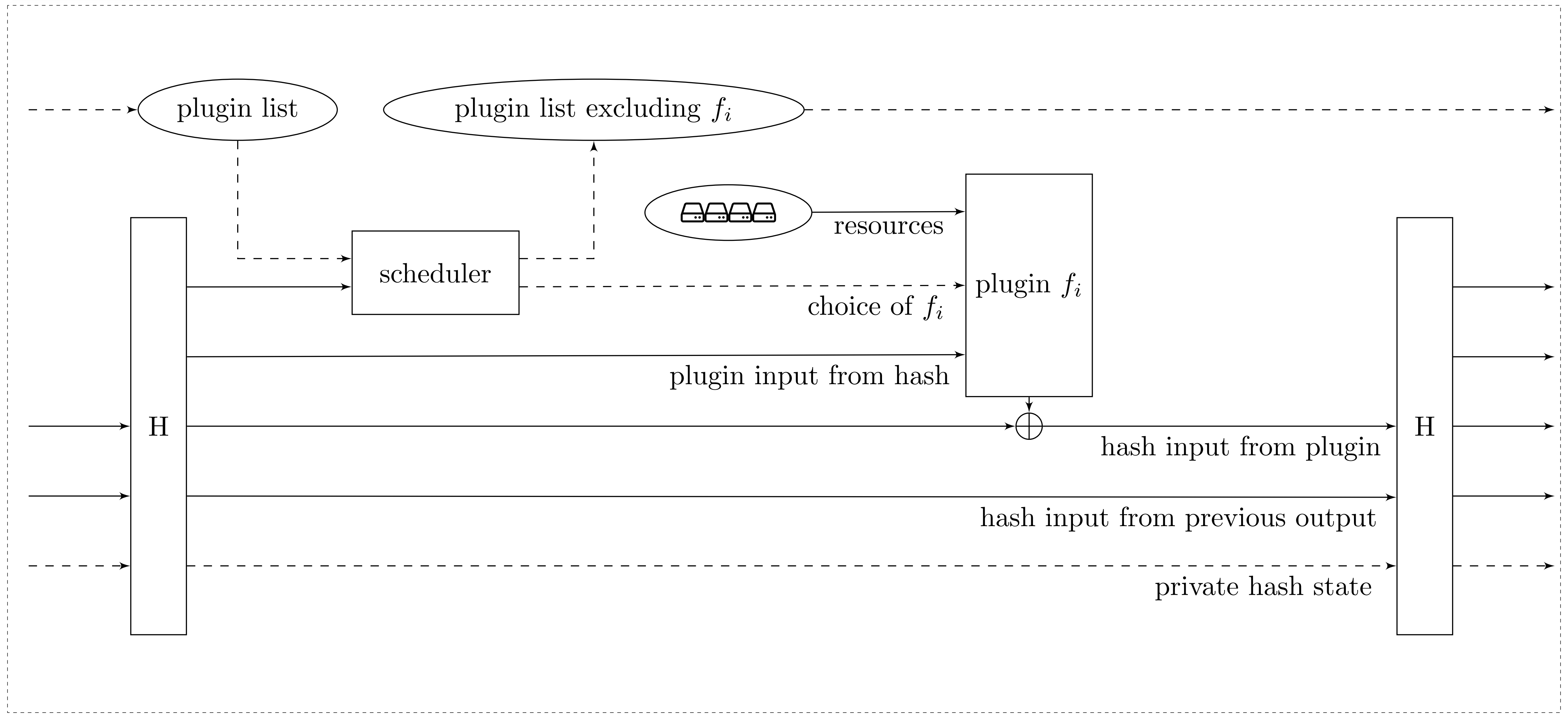
PBKDF Definition



Example Plugins

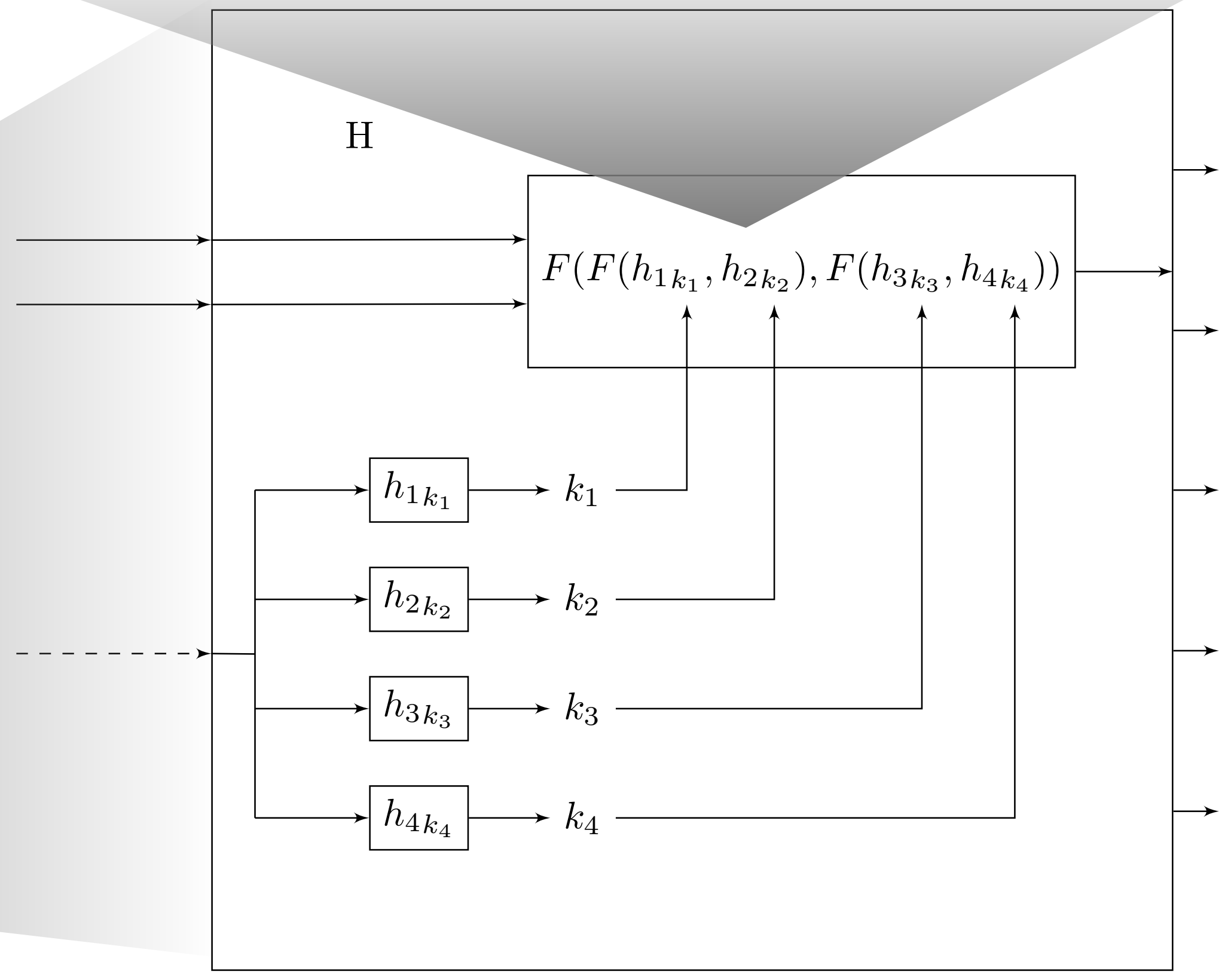
Resource	Plugin
Memory	scrypt[8], argon2d[4]
CPU	Hash functions
Chip/rate limit	TPM
Cache	argon2d[4]
Network	Pythia[6]
. . .	. . .





Example Hash Functions

Hash	Adopted by
SHA2-512 [11]	US/NIST + EU/NESSIE
Whirlpool [9]	Global/ISO + EU/NESSIE
SHA3-512 [12]	US/NIST
Steebog-512 [5]	Russia/FAPSI
BLAKE2-512 [2]	Open source projects
ChaCha20/Poly1305 [7]	Open source projects
AES/Poly1305 [3]	Open source projects
MD6 [10]	Open source projects
. . .	. . .



References

[1] Asic Litecoin/Scrypt Miner Wolf V1 1024 Mh/s (1GH). https://shop.bitmain.com/antminer_l3_litecoin_asic_scrypt_miner.htm.

[2] J.-P. Aumasson, S. Neves, Z. Wilcox-O'Hearn, and C. Winnerlein. BLAKE2: Simpler, smaller, fast as MD5. In M. J. Jacobson Jr., M. E. Locasto, P. Mohassel, and R. Safavi-Naini, editors, *ACNS 13: 11th International Conference on Applied Cryptography and Network Security*, volume 7954 of *Lecture Notes in Computer Science*, pages 119–135, Banff, AB, Canada, June 25–28, 2013. Springer, Heidelberg, Germany.

[3] D. J. Bernstein. The poly1305-AES message-authentication code. In H. Gilbert and H. Handschuh, editors, *Fast Software Encryption – FSE 2005*, volume 3557 of *Lecture Notes in Computer Science*, pages 32–49, Paris, France, Feb. 21–23, 2005. Springer, Heidelberg, Germany.

[4] A. Biryukov, D. Dinu, and D. Khovratovich. Argon2: new generation of memory-hard functions for password hashing and other applications. In *Security and Privacy (EuroS&P), 2016 IEEE European Symposium on*, pages 292–302. IEEE, 2016.

[5] V. Dolmatov and A. Degtyarev. GOST R 34.11-2012: Hash Function. RFC 6986 (Informational), Aug. 2013.

[6] A. Everspaugh, R. Chatterjee, S. Scott, A. Juels, T. Ristenpart, and C. Tech. The pythia prf service. In *USENIX Security Symposium*, pages 547–562, 2015.

[7] A. Langley, W. Chang, N. Mavrogiannopoulos, J. Strombergson, and S. Josefsson. ChaCha20-Poly1305 Cipher Suites for Transport Layer Security (TLS). RFC 7905 (Proposed Standard), June 2016.

[8] C. Percival and S. Josefsson. The scrypt password-based key derivation function. Technical report, 2016.

[9] V. Rijmen and P. S. L. M. Barreto. The WHIRLPOOL hash function.

[10] R. L. Rivest, B. Agre, D. V. Bailey, C. Crutchfield, Y. Dodis, K. E. Fleming, A. Khan, J. Krishnamurthy, Y. Lin, L. Reyzin, et al. The md6 hash function–a proposal to nist for sha-3. *Submission to NIST*, 2(3), 2008.

[11] Secure hash standard. National Institute of Standards and Technology, NIST FIPS PUB 180-2, U.S. Department of Commerce, Aug. 2002.

[12] Secure hash standard. National Institute of Standards and Technology, NIST FIPS PUB 180-4, U.S. Department of Commerce, Aug. 2015.

[13] M. S. Turan, E. Barker, W. Burr, and L. Chen. Recommendation for password-based key derivation. *NIST special publication*, 800:132, 2010.

@BUCompSci