Segurança e arte — animando *hashes*

O problema

A verificação de assinaturas e certificados ocorre por meio da comparação de *hashes* — strings ENORMES e sem sentido, que são difíceis de serem comparadas por seres humanos. Por exemplo, encontre a diferença entre os hashes:

`e0d31d7599daebee65a15a636736d65dae6963d2`
`e0d31d7599daebee65a15a636736d65bae6963d2`

A principal ideia do trabalho é tornar a verificação de conexões HTTPS uma tarefa mais agradável, para que mais usuários verifiquem suas conexões. Isso pode evitar que indivíduos e governos executem ataques MITM, como o governo do Cazaquistão tem tentado desde 2015 (Wikipedia contributors (2019)).

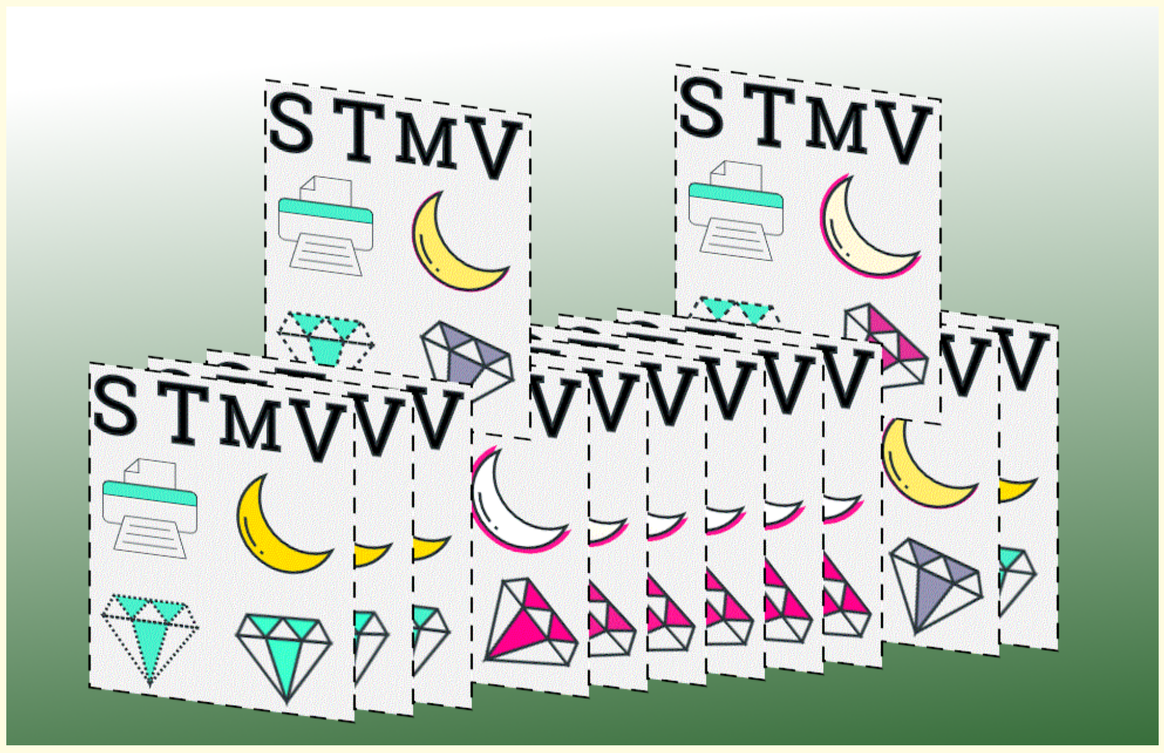
O que foi feito

Para tornar as comparações de hashes mais animadas, foi desenvolvido um protocolo de geração de imagens com animações únicas derivadas a partir de hashes encadeados.

Isso permite a verificação de hashes de maneira rápida e visual — algo que seres humanos fazem melhor.

Esse trabalho expande o esquema desenvolvido por Maina Olembo et al. (2014) adicionando informações no eixo temporal, permitindo menor poluição visual sem que ocorra muita perda de informação.

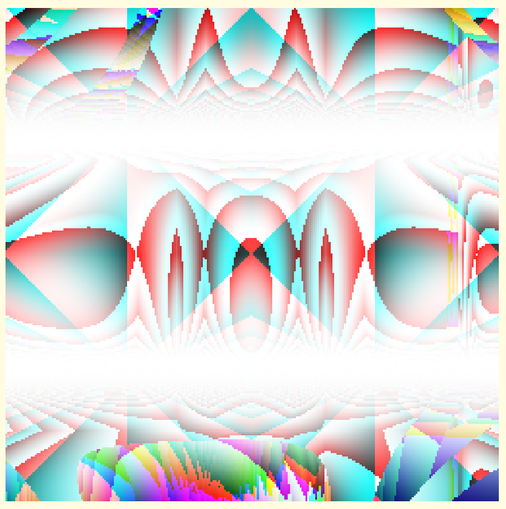
Foi desenvolvido uma biblioteca leve e um protocolo extensível que geram animações utilizando SVG e Javascript. As animações geradas transmitem 52 bits de informação através de letras, ícones e variação de cores e rotação nos ícones.



Quadros de uma animação gerada pela biblioteca desenvolvida

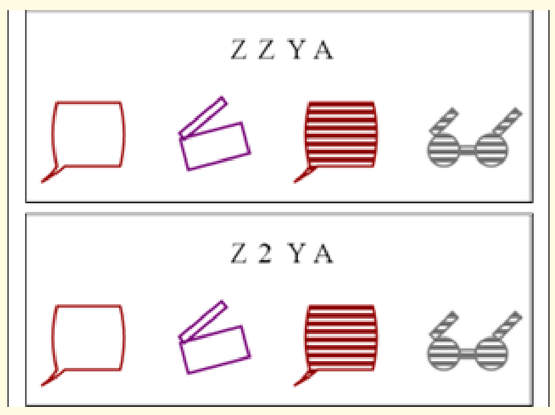
Trabalhos anteriores

A ideia de transformar hashes em imagens tem sido desenvolvida há alguns anos, sendo uma das primeiras tentativas bem sucedidas o desenvolvimento de Random Art por Perrig e Song (1999).



Random art

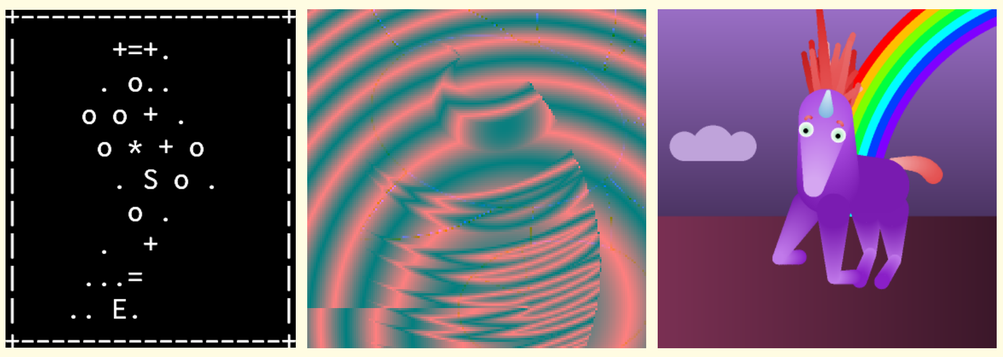
Outro esquema de visualização de hashes foi desenvolvido por Maina Olembo et al. (2014) — o chamado protocolo SCoP, que mistura letras e ícones e é uma das principais bases do trabalho atual.



Duas imagens do protocolo SCoP

Alguns outros esquemas desenvolvidos são Unicorn e Vash (cf. Tan et al. (2017)).

Outro esquema visual é um dos mais usados atualmente: a arte ASCII do OpenSSH.



OpenSSH *Vash* *Unicorn*

Imagens geradas por outros esquemas visuais

Outros usos

Estampas visuais tem outros usos possíveis: estabelecimento de paridade de dispositivos ou até mesmo confirmação visual de que o usuário digitou a senha corretamente — sem vazar nada da senha para terceiros, usando *salts* individuais configurados no navegador.

Verificação da senha digitada

Referências

- M Maina Olembo et al. (2014). "Developing and testing SCoP–a visual hash scheme". Em: *Information Management & Computer Security* 22(4), pgs. 382–392.
- Adrian Perrig e Dawn Song (1999). "Hash Visualization: a New Technique to improve Real-World Security". Em:
- Joshua Tan et al. (2017). "Can Unicorns Help Users Compare Crypto Key Fingerprints?" Em: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. CHI '17. ACM, pgs. 3787–3798.
- Wikipedia contributors (2019). *Kazakhstan man-in-the-middle attack — Wikipedia, The Free Encyclopedia*. [Online; accessado 05/nov/2019]. URL: https://en.wikipedia.org/w/index.php?title=Kazakhstan_man-in-the-middle_attack&oldid=919799991.

Jorge Miguel Ribeiro

jorge.miguel.ribeiro@usp.br

Departamento de Ciências da Computação — Instituto de Matemática e Estatítica — Universidade de São Paulo