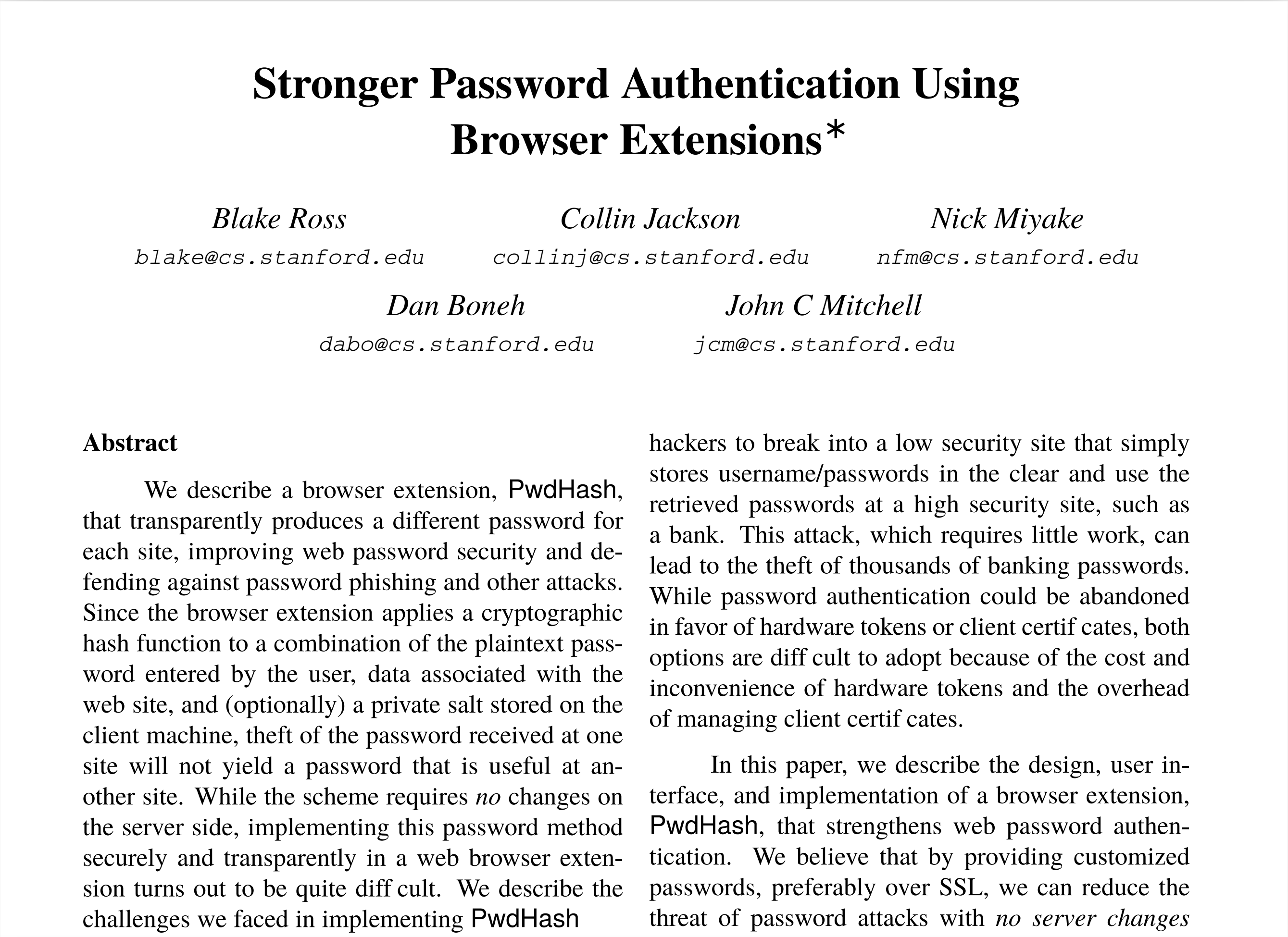

Stronger Password Authentication Using Browser Extensions*

Blake Ross
blake@cs.stanford.edu

Collin Jackson
collinj@cs.stanford.edu

Nick Miyake
nfm@cs.stanford.edu

Dan Boneh
dabo@cs.stanford.edu

John C Mitchell
jcm@cs.stanford.edu

Abstract

We describe a browser extension, PwdHash, that transparently produces a different password for each site, improving web password security and defending against password phishing and other attacks. Since the browser extension applies a cryptographic hash function to a combination of the plaintext password entered by the user, data associated with the web site, and (optionally) a private salt stored on the client machine, theft of the password received at one site will not yield a password that is useful at another site. While the scheme requires *no* changes on the server side, implementing this password method securely and transparently in a web browser extension turns out to be quite difficult. We describe the challenges we faced in implementing PwdHash

hackers to break into a low security site that simply stores username/passwords in the clear and use the retrieved passwords at a high security site, such as a bank. This attack, which requires little work, can lead to the theft of thousands of banking passwords. While password authentication could be abandoned in favor of hardware tokens or client certificates, both options are difficult to adopt because of the cost and inconvenience of hardware tokens and the overhead of managing client certificates.

In this paper, we describe the design, user interface, and implementation of a browser extension, PwdHash, that strengthens web password authentication. We believe that by providing customized passwords, preferably over SSL, we can reduce the threat of password attacks with *no server changes*