# Requirements for Applications Using an ESAPI Implementation

This set of requirements define the expected behavior of applications that use an ESAPI implementation for security. Although an ESAPI implementation provides extensive security controls, the methods must be used properly in order to ensure a secure application.

# Authentication

## Call Authenticator.getInstance().login() on every request

*Rationale: Calling login will set the current user for that thread of execution. This enables logging, access control, and much more. You should put the call to login in a strategic location where it will be very difficult to avoid. A filter or controller are good choices.*

## Call Authenticator.getInstance().logout()

*Rationale: Calling logout…*

## Put the user’s last login time on the successful login page

*Rationale: Calling logout…*

## Put the user’s last login time on the successful login page

*Rationale: Calling logout…*

## Put the user’s last login time on the successful login page

*Rationale: Calling logout…*

# Access Control

## Call Authenticator.getInstance().login() on every request

*Rationale: Calling …*

# Requirements for ESAPI Implementations

This set of requirements define the expected behavior of ESAPI implementations. These requirements are intended to apply only to people actually implementing the ESAPI interfaces. This document will serve as a guideline to ensure that implementations claiming ESAPI conformance have actually implemented the ESAPI successfully.

# Authentication

## Asdf

Rationale: this prevents…

## Test

# Access Control

## Test