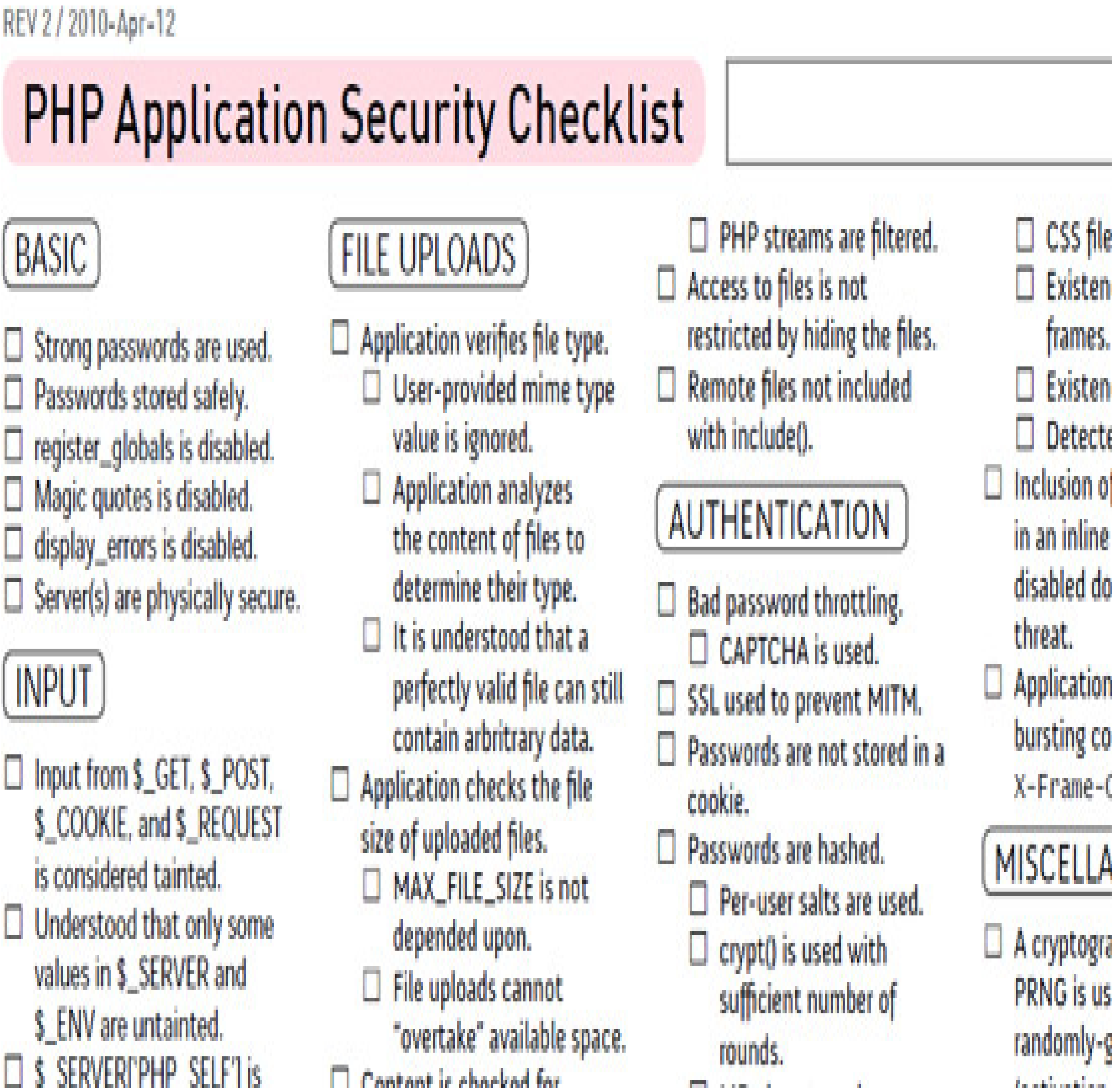

REV 2 / 2010-Apr-12

# PHP Application Security Checklist

## BASIC

- □ Strong passwords are used.
- □ Passwords stored safely.
- □ register_globals is disabled.
- □ Magic quotes is disabled.
- □ display_errors is disabled.
- □ Server(s) are physically secure.

## INPUT

- □ Input from $_GET, $_POST, $_COOKIE, and $_REQUEST is considered tainted.
- □ Understood that only some values in $_SERVER and $_ENV are untainted.
- □ $_SERVER['PHP_SELF'] is

## FILE UPLOADS

- □ Application verifies file type.
  - □ User-provided mime type value is ignored.
  - □ Application analyzes the content of files to determine their type.
  - □ It is understood that a perfectly valid file can still contain arbitrary data.
- □ Application checks the file size of uploaded files.
  - □ MAX_FILE_SIZE is not depended upon.
  - □ File uploads cannot "overtake" available space.
- □ Content is checked for

  - □ PHP streams are filtered.
- □ Access to files is not restricted by hiding the files.
- □ Remote files not included with include().

## AUTHENTICATION

- □ Bad password throttling.
  - □ CAPTCHA is used.
- □ SSL used to prevent MITM.
- □ Passwords are not stored in a cookie.
- □ Passwords are hashed.
  - □ Per-user salts are used.
  - □ crypt() is used with sufficient number of rounds.

  - □ CSS file
  - □ Existen frames.
  - □ Existen
  - □ Detecte
- □ Inclusion o in an inline disabled do threat.
- □ Application bursting co X-Frame-(

## MISCELLA

- □ A cryptogra PRNG is us randomly-g (activation