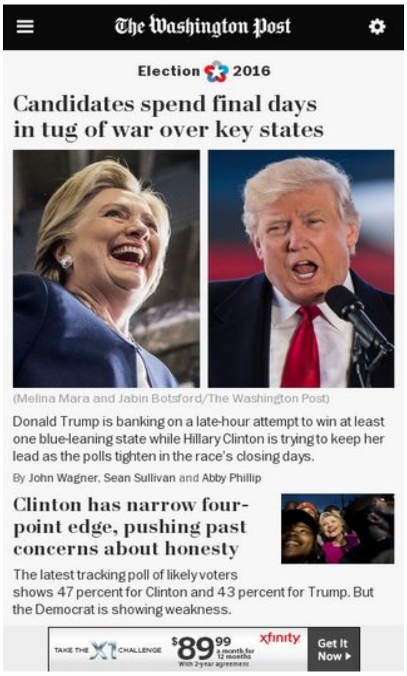
PageSpeed Insights

Mobile



The Washington Post

Election 2016

Candidates spend final days in tug of war over key states

(Melina Mara and Jabin Botsford/The Washington Post)

Donald Trump is banking on a late-hour attempt to win at least one blue-leaning state while Hillary Clinton is trying to keep her lead as the polls tighten in the race's closing days.

By John Wagner, Sean Sullivan and Abby Phillip

Clinton has narrow four-point edge, pushing past concerns about honesty

The latest tracking poll of likely voters shows 47 percent for Clinton and 43 percent for Trump. But the Democrat is showing weakness.

58 / 100 Speed

Should Fix:

Leverage browser caching

Setting an expiry date or a maximum age in the HTTP headers for static resources instructs the browser to load previously downloaded resources from local disk rather than over the network.

Leverage browser caching for the following cacheable resources:

- https://cdn-api.arcpublishing.com/v1.0/loxodo/datapoint/save (expiration not specified)
- https://css.washingtonpost.com/wp-srv/ad/public/brandconnect/modulev3/images/glyphicons_195_circle_info_grey.png (expiration not specified)
- https://css.washingtonpost.com/wp-stat/ad/loaders/latest/css/wp_mobile.css (expiration not specified)
- https://d21rhj7n383afu.cloudfront.net/JWPlayer/JWPlayerConfig.js (expiration not specified)
- https://d21rhj7n383afu.cloudfront.net/WapoVideo/WapoVideoConfig.js (expiration not

specified)

- https://js.washingtonpost.com/wp-stat/rum/wprum.min.js (expiration not specified)
- https://loxodo-ct.ext.nile.works/api/save (expiration not specified)
- https://www.washingtonpost.com/wp-stat/ad/loaders/latest/js/min/loader.min.js (expiration not specified)
- https://js.washingtonpost.com/wp-stat/advertising/pseudo-static/vidExclusions/vid_exclusions.js (5 minutes)
- https://js.washingtonpost.com/wp-stat/analytics/main.js (5 minutes)
- https://www.washingtonpost.com/wp-stat/advertising/BrandConnect/js/min/scripts-1.0.0.min.js (5 minutes)
- https://www.washingtonpost.com/wp-stat/advertising/pseudo-static/identity-retargeting.js (5 minutes)
- https://js.washingtonpost.com/video/resources/env/WapoVideo/css/WapoVideoPlayer.min.css?_=16f2672 (10 minutes)
- https://js.washingtonpost.com/video/resources/env/WapoVideo/dist/WapoVideoEmbed.main.min.js?_=16f2672 (10 minutes)
- https://js.washingtonpost.com/video/resources/env/WapoVideo/dist/WapoVideoEmbed.min.js (10 minutes)
- https://js.washingtonpost.com/video/resources/js/posttv/jwPlayer/jwPlayerWrapper.min.js?_=16f2672 (10 minutes)
- https://js.washingtonpost.com/video/resources/js/posttv/vendor/es6-promise.min.js (10 minutes)
- https://js.washingtonpost.com/video/resources/js/posttv/vendor/jw/jwplayer.js?_=16f2672 (10 minutes)
- https://js.washingtonpost.com/video/resources/js/posttv/vendor/pubsub.min.js (10 minutes)
- https://js.washingtonpost.com/video/resources/js/posttv/vendor/streamsense.min.js (10 minutes)
- https://js.washingtonpost.com/video/resources/js/posttv/vendor/underscore-min.js (10 minutes)
- https://js-sec.indexww.com/ht/htw-wapo.js (23.2 minutes)
- https://s-jsonp.moatads.com/ocr/WASHPOSTCW1/level3/57952378-57952378?t=201610538 (26.6 minutes)
- https://pagead2.googlesyndication.com/pagead/osd.js (60 minutes)
- https://washingtonpost-d.openx.net/w/1.0/jstag?nc=701-WashPo_Mobile (60 minutes)
- https://z.moatads.com/washpostdfpcw45893854/moatad.js (9.4 hours)

Mobile

Eliminate render-blocking JavaScript and CSS in above-the-fold content

Your page has 11 blocking script resources and 5 blocking CSS resources. This causes a delay in rendering your page.

None of the above-the-fold content on your page could be rendered without waiting for the following resources to load. Try to defer or asynchronously load blocking resources, or inline the critical portions of those resources directly in the HTML.

Remove render-blocking JavaScript:

- https://www.washingtonpost.com/pb/gr/c/mobile-homepage/r0CzmrNiXVXF1q/load_immediately/f65e4f3f81.js?_=cf22c
- https://www.washingtonpost.com/pb/resources/js/head.min.js
- https://www.washingtonpost.com/pb/gr/c/mobile-homepage/r0CzmrNiXVXF1q/global/9adf0167ec.js?_=0b3d0
- https://www.washingtonpost.com/pb/gr/c/mobile-homepage/r0CzmrNiXVXF1q/headjs/16db91d4b3.js?_=3f62d
- https://www.washingtonpost.com/pb/gr/p/mobile-homepage/r0CzmrNiXVXF1q/head.js?_=55491
- https://www.washingtonpost.com/pb/gr/c/mobile-homepage/r0CzmrNiXVXF1q/hi-pri-js/e7f066f51e.js?_=3f62d
- https://www.washingtonpost.com/pb/gr/p/mobile-homepage/r0CzmrNiXVXF1q/hi-pri-render.js?_=55491
- https://www.washingtonpost.com/pb/gr/c/mobile-homepage/r0CzmrNiXVXF1q/js/fdfbc5dc06.js?_=fcc8d
- https://www.washingtonpost.com/pb/gr/p/mobile-homepage/r0CzmrNiXVXF1q/render.js?_=55491
- https://www.washingtonpost.com/pb/gr/p/mobile-homepage/r0CzmrNiXVXF1q/instance.js?_=55491
- https://www.washingtonpost.com/pb/gr/c/mobile-homepage/r0CzmrNiXVXF1q/after_features/ab2a22c20d.js?_=077b3

Optimize CSS Delivery of the following:

- https://www.washingtonpost.com/pb/resources/css/latest/headerfonts.latest.css?_=2016110

Mobile

31211PST

- https://www.washingtonpost.com/pb/gr/c/mobile-homepage/r0CzmrNiXVXF1q/css/5b50915464.css?_=24923
- https://www.washingtonpost.com/pb/gr/p/mobile-homepage/r0CzmrNiXVXF1q/style.css?_=55491
- https://www.washingtonpost.com/pb/gr/c/mobile-homepage/r0CzmrNiXVXF1q/modal-css/938c258067.css?_=00223
- https://css.washingtonpost.com/wp-stat/ad/loaders/latest/css/wp_mobile.css

Consider Fixing:

Enable compression

Compressing resources with gzip or deflate can reduce the number of bytes sent over the network.

Enable compression for the following resources to reduce their transfer size by 7.5KiB (72% reduction).

- Compressing https://mongo-good-server.ext.nile.works/test-statistics?q=%7B%22_id%22:%7B%22$in%22:%5B%220fqoQodF1q3-c%22%5D%7D%7D could save 5.8KiB (76% reduction).
- Compressing https://d21rhj7n383afu.cloudfront.net/JWPlayer/JWPlayerConfig.js could save 1.1KiB (62% reduction).
- Compressing https://d21rhj7n383afu.cloudfront.net/WapoVideo/WapoVideoConfig.js could save 577B (57% reduction).

Minify JavaScript

Compacting JavaScript code can save many bytes of data and speed up downloading, parsing, and execution time.

Minify JavaScript for the following resources to reduce their size by 7.8KiB (19% reduction).

- Minifying https://js.washingtonpost.com/wp-stat/analytics/main.js could save 6.7KiB (17% reduction) after compression.
- Minifying https://d21rhj7n383afu.cloudfront.net/WapoVideo/WapoVideoConfig.js could save 613B (60% reduction).
- Minifying https://d21rhj7n383afu.cloudfront.net/JWPlayer/JWPlayerConfig.js could save 597B (34% reduction).

Optimize images

Properly formatting and compressing images can save many bytes of data.

Optimize the following images to reduce their size by 10.5KiB (25% reduction).

- Compressing https://www.washingtonpost.com/pb/resources/img/thewashingtonpost-white-2x.png could save 3.4KiB (28% reduction).
- Compressing https://img.washingtonpost.com/wp-apps/imrs.php?src=https://img.washingtonpost.com/blogs/the-fix/files/2016/11/Screen-Shot-2016-11-04-at-12.16.15-PM.png&w=138&h=92 could save 3.1KiB (20% reduction).
- Compressing https://css.washingtonpost.com/wp-srv/ad/public/brandconnect/modulev3/images/glyphicons_195_circle_info_grey.png could save 1.2KiB (80% reduction).
- Compressing https://www.washingtonpost.com/rf/image_48h/2010-2019/WashingtonPost/2016/09/30/Interactivity/Images/Campaign2016Star-24px-high.png could save 1.2KiB (38% reduction).
- Compressing https://tpc.googlesyndication.com/pagead/imgad?id=CICAgKDLtpPLtAEQARgBMgiBV8eHj8LEgQ could save 850B (11% reduction).
- Compressing https://www.washingtonpost.com/pb/resources/img/washingtonpost_black_32.png could save 724B (34% reduction).

✓ 5 Passed Rules

Avoid landing page redirects

Mobile

Your page has no redirects. Learn more about avoiding landing page redirects.

Reduce server response time

Your server responded quickly. Learn more about server response time optimization.

Minify CSS

Your CSS is minified. Learn more about minifying CSS.

Minify HTML

Your HTML is minified. Learn more about minifying HTML.

Prioritize visible content

You have the above-the-fold content properly prioritized. Learn more about prioritizing visible content.

99 / 100 User Experience

Consider Fixing:

Size tap targets appropriately

Some of the links/buttons on your webpage may be too small for a user to easily tap on a touchscreen. Consider making these tap targets larger to provide a better user experience.

The following tap targets are close to other nearby tap targets and may need additional spacing around them.

- The tap target <a href="https://www.washingtonpost.com" class="wp-logo-link"></a> is close to 2 other tap targets final.

- The tap target <a href="#" class="section-menu-btn"></a> is close to 2 other tap targets final.
- The tap target <a id="settings-nav-btn" href="#"></a> is close to 2 other tap targets final.
- The tap target <a href="https://www.wa…bb4_story.html"></a> is close to 1 other tap targets final.
- The tap target <a href="https://www.wa…bb4_story.html" class="">Chris Christie…e’ convictions</a> and 3 others are close to other tap targets final.
- The tap target <a href="http://www.was…le/ruth-marcus">Ruth Marcus</a> and 4 others are close to other tap targets.
- The tap target <a href="https://www.wa…865_story.html" class="related">Putin struggle…’ for Russians</a> and 3 others are close to other tap targets.

✓ 4 Passed Rules

Avoid plugins

Your page does not appear to use plugins, which would prevent content from being usable on many platforms. Learn more about the importance of avoiding plugins.

Configure the viewport

Your page specifies a viewport matching the device's size, which allows it to render properly on all devices. Learn more about configuring viewports.

Size content to viewport

The contents of your page fit within the viewport. Learn more about sizing content to the viewport.

Use legible font sizes

The text on your page is legible. Learn more about using legible font sizes.

Desktop

The Washington Post

Candidates spend final days in tug of war over key states

Clinton has narrow four-point edge, pushing past concerns about honesty

The electoral map is definitely moving in Trump's direction

In Miami's 'Little Moscow,' Russians behind Trump's

56 / 100 Speed

Should Fix:

Leverage browser caching

Setting an expiry date or a maximum age in the HTTP headers for static resources instructs the browser to load previously downloaded resources from local disk rather than over the network.

Leverage browser caching for the following cacheable resources:

- https://cdn-api.arcpublishing.com/v1.0/loxodo/datapoint/save (expiration not specified)
- https://css.washingtonpost.com/wp-srv/ad/public/brandconnect/modulev3/images/glyphicons_195_circle_info_grey.png (expiration not specified)
- https://css.washingtonpost.com/wp-stat/ad/loaders/latest/css/wp.css (expiration not specified)
- https://d21rhj7n383afu.cloudfront.net/JWPlayer/JWPlayerConfig.js (expiration not specified)
- https://d21rhj7n383afu.cloudfront.net/WapoVideo/WapoVideoConfig.js (expiration not specified)
- https://img.washingtonpost.com/wp-srv/hp/cube.png (expiration not specified)
- https://js.washingtonpost.com/wp-stat/rum/wprum.min.js (expiration not specified)
- https://loxodo-ct.ext.nile.works/api/save (expiration not specified)
- https://www.washingtonpost.com/wp-stat/ad/loaders/latest/js/min/loader.min.js (expiration not specified)
- https://www.washingtonpost.com/pb/api/v2/render/feature/variant/0fEjgFtF1qL?rid=rl0ZZk2vXVXF1q&uri=/homepage/&outputType=default (2 minutes)

Desktop

- https://img.washingtonpost.com/rw/WashingtonPost/Content/Epaper/2016-11-05/Ax1_module2.jpg (5 minutes)
- https://js.washingtonpost.com/wp-stat/advertising/pseudo-static/vidExclusions/vid_exclusions.js (5 minutes)
- https://js.washingtonpost.com/wp-stat/analytics/main.js (5 minutes)
- https://www.washingtonpost.com/r/2010-2019/WashingtonPost/2016/10/19/Style/Images/101916-Microsoft-Logo.png (5 minutes)
- https://www.washingtonpost.com/wp-stat/advertising/BrandConnect/js/min/scripts-1.0.0.min.js (5 minutes)
- https://www.washingtonpost.com/wp-stat/advertising/pseudo-static/identity-retargeting.js (5 minutes)
- https://www.washingtonpost.com/wp-stat/advertising/spacer.gif (5 minutes)
- https://js.washingtonpost.com/video/resources/env/WapoVideo/css/WapoVideoPlayer.min.css?_=16f2672 (10 minutes)
- https://js.washingtonpost.com/video/resources/env/WapoVideo/dist/WapoVideoEmbed.main.min.js?_=16f2672 (10 minutes)
- https://js.washingtonpost.com/video/resources/env/WapoVideo/dist/WapoVideoEmbed.min.js (10 minutes)
- https://js.washingtonpost.com/video/resources/js/posttv/jwPlayer/jwPlayerWrapper.min.js?_=16f2672 (10 minutes)
- https://js.washingtonpost.com/video/resources/js/posttv/vendor/es6-promise.min.js (10 minutes)
- https://js.washingtonpost.com/video/resources/js/posttv/vendor/jw/jwplayer.js?_=16f2672 (10 minutes)
- https://js.washingtonpost.com/video/resources/js/posttv/vendor/pubsub.min.js (10 minutes)
- https://js.washingtonpost.com/video/resources/js/posttv/vendor/streamsense.min.js (10 minutes)
- https://js.washingtonpost.com/video/resources/js/posttv/vendor/underscore-min.js (10 minutes)
- https://js-sec.indexww.com/ht/htw-wapo.js (23.1 minutes)
- https://pagead2.googlesyndication.com/pagead/osd.js (60 minutes)
- https://washingtonpost-d.openx.net/w/1.0/jstag?nc=701-Washingtonpost (60 minutes)
- https://z.moatads.com/washpostdfpcw45893854/moatad.js (9.4 hours)

Desktop

Eliminate render-blocking JavaScript and CSS in above-the-fold content

Your page has 5 blocking script resources and 4 blocking CSS resources. This causes a delay in rendering your page.

None of the above-the-fold content on your page could be rendered without waiting for the following resources to load. Try to defer or asynchronously load blocking resources, or inline the critical portions of those resources directly in the HTML.

Remove render-blocking JavaScript:

- https://www.washingtonpost.com/pb/gr/c/default/rl0ZZk2vXVXF1q/load_immediately/f65e4f3f81.js?_=cf22c
- https://www.washingtonpost.com/pb/resources/js/head.min.js
- https://www.washingtonpost.com/pb/gr/c/default/rl0ZZk2vXVXF1q/global/9adf0167ec.js?_=0b3d0
- https://www.washingtonpost.com/pb/gr/c/default/rl0ZZk2vXVXF1q/headjs/70b9918770.js?_=a857b
- https://www.washingtonpost.com/pb/gr/p/default/rl0ZZk2vXVXF1q/head.js?_=f6421

Optimize CSS Delivery of the following:

- https://www.washingtonpost.com/pb/resources/css/latest/headerfonts.latest.css?_=201611031211PST
- https://www.washingtonpost.com/pb/gr/c/default/rl0ZZk2vXVXF1q/css/9e1acbfb3f.css?_=650f9
- https://www.washingtonpost.com/pb/gr/p/default/rl0ZZk2vXVXF1q/style.css?_=f6421
- https://www.washingtonpost.com/pb/gr/c/default/rl0ZZk2vXVXF1q/modal-css/938c258067.css?_=00223

Optimize images

Properly formatting and compressing images can save many bytes of data.

Desktop

Optimize the following images to reduce their size by 526.6KiB (82% reduction).

- Compressing and resizing https://img.washingtonpost.com/rf/image_992x661/2010-2019/WashingtonPost/2016/10/25/Others/Images/2016-10-25/TRUMPRUSSIANS25_111477432866.jpg could save 202.6KiB (98% reduction).
- Compressing and resizing https://img.washingtonpost.com/rf/image_1248w/2010-2019/WashingtonPost/2016/11/04/Interactivity/Images/fix-predictions-hpdisplay.jpg could save 145.1KiB (89% reduction).
- Compressing and resizing https://img.washingtonpost.com/rf/image_480x320/2010-2019/WashingtonPost/2016/11/05/Interactivity/Images/yulbrynner.jpg could save 39.2KiB (84% reduction).
- Compressing and resizing https://img.washingtonpost.com/wp-apps/imrs.php?src=https://tpc.googlesyndication.com/pagead/imgad?id=CICAgKDLttPq_gEQARgBMgiKn27EenvZRQ&w=400&h=300 could save 37.6KiB (84% reduction).
- Compressing and resizing https://img.washingtonpost.com/rf/image_480x480/2010-2019/WashingtonPost/2016/11/03/Others/Images/2016-11-03/hrc401478216172.jpg could save 34.3KiB (96% reduction).
- Compressing and resizing https://img.washingtonpost.com/rw/WashingtonPost/Content/Epaper/2016-11-05/Ax1_module2.jpg could save 16.7KiB (83% reduction).
- Compressing and resizing https://www.washingtonpost.com/pb/resources/img/thewashingtonpost-white-2x.png could save 9.2KiB (75% reduction).
- Compressing https://img.washingtonpost.com/wp-apps/imrs.php?src=http%3A%2F%2Fwww.washingtonpost.com%2Fpb%2Fresources%2Fimg%2Fposttv%2FWP_Grey.jpg&w=1280&h=720 could save 7.2KiB (18% reduction).
- Compressing and resizing https://img.washingtonpost.com/wp-apps/imrs.php?src=http://img.washingtonpost.com/news/global-opinions/wp-content/uploads/sites/59/2016/11/AFP_HN6AW.jpg&w=112&h=112 could save 5.8KiB (81% reduction).
- Compressing and resizing https://img.washingtonpost.com/rf/image_112x112/2010-2019/WashingtonPost/2016/11/04/Editorial-Opinion/Images/AFP_HS6AI.jpg could save 4.5KiB (77% reduction).
- Compressing and resizing https://img.washingtonpost.com/rf/image_112x112/2010-2019/WashingtonPost/2016/11/04/Editorial-Opinion/Images/2013000.jpg could save 3.8KiB (77% reduction).
- Compressing and resizing https://img.washingtonpost.com/rf/image_112x112/2010-2019/WashingtonPost/2016/11/04/Editorial-Opinion/Images/Campaign_2016_Clinton_Email_Comey-27a40.jpg could save 3.6KiB (76% reduction).
- Compressing and resizing https://img.washingtonpost.com/rf/image_112x112/2010-2019/WashingtonPost/2016/11/03/National-Politics/Images/ht_41474939211.jpg could save 3.3KiB (75% reduction).
- Compressing https://img.washingtonpost.com/wp-apps/imrs.php?src=https://img.washington

post.com/blogs/the-fix/files/2016/11/Screen-Shot-2016-11-04-at-12.16.15-PM.png&w=138&h=92 could save 3.1KiB (20% reduction).

- Compressing and resizing https://img.washingtonpost.com/rf/image_112x112/2010-2019/WashingtonPost/2016/11/04/Editorial-Opinion/Images/621091020.jpg could save 3KiB (75% reduction).
- Compressing and resizing https://www.washingtonpost.com/rf/image_48h/2010-2019/WashingtonPost/2016/09/30/Interactivity/Images/Campaign2016Star-24px-high.png could save 1.9KiB (59% reduction).
- Compressing https://css.washingtonpost.com/wp-srv/ad/public/brandconnect/modulev3/images/glyphicons_195_circle_info_grey.png could save 1.2KiB (80% reduction).
- Compressing https://img.washingtonpost.com/pb/resources/img/sprites/ad-sprite.png?t=20130227155400 could save 1.1KiB (53% reduction).
- Compressing https://img.washingtonpost.com/wp-apps/imrs.php?src=https://img.washingtonpost.com/blogs/the-fix/files/2016/11/Screen-Shot-2016-11-04-at-12.16.15-PM.png&w=60&h=60 could save 1,020B (19% reduction).
- Compressing and resizing https://www.washingtonpost.com/r/2010-2019/WashingtonPost/2016/10/19/Style/Images/101916-Microsoft-Logo.png could save 900B (26% reduction).
- Compressing https://img.washingtonpost.com/wp-srv/hp/cube.png could save 851B (15% reduction).
- Compressing https://www.washingtonpost.com/pb/resources/img/washingtonpost_black_32.png could save 724B (34% reduction).

Consider Fixing:

Enable compression

Compressing resources with gzip or deflate can reduce the number of bytes sent over the network.

Enable compression for the following resources to reduce their transfer size by 7.4KiB (72% reduction).

- Compressing https://mongo-good-server.ext.nile.works/test-statistics?q=%7B%22_id%22:%7B%22$in%22:%5B%220fqoQodF1q3%22%5D%7D%7D could save 5.8KiB (76% reduction).
- Compressing https://d21rhj7n383afu.cloudfront.net/JWPlayer/JWPlayerConfig.js could save

1.1KiB (62% reduction).

- Compressing https://d21rhj7n383afu.cloudfront.net/WapoVideo/WapoVideoConfig.js could save 577B (57% reduction).

Minify JavaScript

Compacting JavaScript code can save many bytes of data and speed up downloading, parsing, and execution time.

Minify JavaScript for the following resources to reduce their size by 7.8KiB (19% reduction).

- Minifying https://js.washingtonpost.com/wp-stat/analytics/main.js could save 6.7KiB (17% reduction) after compression.
- Minifying https://d21rhj7n383afu.cloudfront.net/WapoVideo/WapoVideoConfig.js could save 613B (60% reduction).
- Minifying https://d21rhj7n383afu.cloudfront.net/JWPlayer/JWPlayerConfig.js could save 597B (34% reduction).

✓ 5 Passed Rules

Avoid landing page redirects

Your page has no redirects. Learn more about avoiding landing page redirects.

Reduce server response time

Your server responded quickly. Learn more about server response time optimization.

Minify CSS

Your CSS is minified. Learn more about minifying CSS.

Desktop

Minify HTML

Your HTML is minified. Learn more about minifying HTML.

Prioritize visible content

You have the above-the-fold content properly prioritized. Learn more about prioritizing visible content.