Item 103
git_comments:

git_commits:

1. **summary:** Zookeeper version is updated. (#8363)
   message: Zookeeper version is updated. (#8363) * Zookeeper version is updated. * Zookeeper version is updated at licenses.yaml * licenses.yaml is updated and dependencies are fixed to make the project successfully build. * Zookeeper versions are fixed at licenses.yaml

github_issues:

1. **title:** Upgrade Zookeper Version to 3.4.14
   body: Zookeeper version should be upgraded to version 3.4.14 or later due to it has a [vulnerability](https://nvd.nist.gov/vuln/detail/CVE-2019-0201): > An issue is present in Apache ZooKeeper 1.0.0 to 3.4.13 and 3.5.0-alpha to 3.5.4-beta. ZooKeeper's getACL() command doesn't check any permission when retrieves the ACLs of the requested node and returns all information contained in the ACL Id field as plaintext string. DigestAuthenticationProvider overloads the Id field with the hash value that is used for user authentication. As a consequence, if Digest Authentication is in use, the unsalted hash value will be disclosed by getACL() request for unauthenticated or unprivileged users. and has a critical [bug](https://issues.apache.org/jira/browse/ZOOKEEPER-2960).
2. **title:** Upgrade Zookeper Version to 3.4.14
   body: Zookeeper version should be upgraded to version 3.4.14 or later due to it has a [vulnerability](https://nvd.nist.gov/vuln/detail/CVE-2019-0201): > An issue is present in Apache ZooKeeper 1.0.0 to 3.4.13 and 3.5.0-alpha to 3.5.4-beta. ZooKeeper's getACL() command doesn't check any permission when retrieves the ACLs of the requested node and returns all information contained in the ACL Id field as plaintext string. DigestAuthenticationProvider overloads the Id field with the hash value that is used for user authentication. As a consequence, if Digest Authentication is in use, the unsalted hash value will be disclosed by getACL() request for unauthenticated or unprivileged users. and has a critical [bug](https://issues.apache.org/jira/browse/ZOOKEEPER-2960).

github_issues_comments:

github_pulls:

1. **title:** Zookeeper version is updated.
   body: <!-- Thanks for trying to help us make Apache Druid (incubating) be the best it can be! Please fill out as much of the following information as is possible (where relevant, and remove it when irrelevant) to help make the intention and scope of this PR clear in order to ease review. --> Fixes #8362. <!-- Replace XXXX with the id of the issue fixed in this PR. Remove this section if there is no corresponding issue. Don't reference the issue in the title of this pull-request. --> <!-- If you are a committer, follow the PR action item checklist for committers: https://github.com/apache/incubator-druid/blob/master/dev/committer-instructions.md#pr-and-issue-action-item-checklist-for-committers. --> ### Description <!-- Describe the goal of this PR, what problem are you fixing. If there is a corresponding issue (referenced above), it's not necessary to repeat the description here, however, you may choose to keep one summary sentence. --> Zookeeper version should be upgraded to version 3.4.14 or later due to it has a [vulnerability](https://nvd.nist.gov/vuln/detail/CVE-2019-0201): > An issue is present in Apache ZooKeeper 1.0.0 to 3.4.13 and 3.5.0-alpha to 3.5.4-beta. ZooKeeper's getACL() command doesn't check any permission when retrieves the ACLs of the requested node and returns all information contained in the ACL Id field as plaintext string. DigestAuthenticationProvider overloads the Id field with the hash value that is used for user authentication. As a consequence, if Digest Authentication is in use, the unsalted hash value will be disclosed by getACL() request for unauthenticated or unprivileged users. and has a critical [bug](https://issues.apache.org/jira/browse/ZOOKEEPER-2960). <!-- Describe your patch: what did you change in code? How did you fix the problem? --> Zookeeper version is upgraded to 3.4.14 <!-- If there are several relatively logically separate changes in this PR, create a mini-section for each of them. For example: --> <!-- In each section, please describe design decisions made, including: - Choice of algorithms - Behavioral aspects. What configuration values are acceptable? How are corner cases and error conditions handled, such as when there are insufficient resources? - Class organization and design (how the logic is split between classes, inheritance, composition, design patterns) - Method organization and design (how the logic is split between methods, parameters and return types) - Naming (class, method, API, configuration, HTTP endpoint, names of emitted metrics) --> <!-- It's good to describe an alternative design (or mention an alternative name) for every design (or naming) decision point and compare the alternatives with the designs that you've implemented (or the names you've chosen) to highlight the advantages of the chosen designs and names. --> <!-- If there was a discussion of the design of the feature implemented in this PR elsewhere (e. g. a "Proposal" issue, any other issue, or a thread in the development mailing list), link to that discussion from this PR description and explain what have changed in your final design compared to your original proposal or the consensus version in the end of the discussion. If something hasn't changed since the original discussion, you can omit a detailed discussion of those aspects of the design here, perhaps apart from brief mentioning for the sake of readability of this PR description. --> <!-- Some of the aspects mentioned above may be omitted for simple and small changes. --> <hr> This PR has: - [X] been self-reviewed.<!-- - [ ] using the [concurrency checklist](https://github.com/apache/incubator-druid/blob/master/dev/code-review/concurrency.md) (Remove this item if the PR doesn't have any relation to concurrency.) --> - [ ] added documentation for new or modified features or behaviors. - [ ] added Javadocs for most classes and all non-trivial methods. Linked related entities via Javadoc links. - [ ] added comments explaining the "why" and the intent of the code wherever would not be obvious for an unfamiliar reader. - [ ] added unit tests or modified existing tests to cover new code paths. - [ ] added integration tests. - [ ] been tested in a test Druid cluster. <!-- Check the items by putting "x" in the brackets for the done things. Not all of these items apply to every PR. Remove the items which are not done or not relevant to the PR. None of the items from the checklist above are strictly necessary, but it would be very helpful if you at least self-review the PR. --> <hr>
2. **title:** Zookeeper version is updated.
   body: <!-- Thanks for trying to help us make Apache Druid (incubating) be the best it can be! Please fill out as much of the following information as is possible (where relevant, and remove it when irrelevant) to help make the intention and scope of this PR clear in order to ease review. --> Fixes #8362. <!-- Replace XXXX with the id of the issue fixed in this PR. Remove this section if there is no corresponding issue. Don't reference the issue in the title of this pull-request. --> <!-- If you are a committer, follow the PR action item checklist for committers: https://github.com/apache/incubator-druid/blob/master/dev/committer-instructions.md#pr-and-issue-action-item-checklist-for-committers. --> ### Description <!-- Describe the goal of this PR, what problem are you fixing. If there is a corresponding issue (referenced above), it's not necessary to repeat the description here, however, you may choose to keep one summary sentence. --> Zookeeper version should be upgraded to version 3.4.14 or later due to it has a [vulnerability](https://nvd.nist.gov/vuln/detail/CVE-2019-0201): > An issue is present in Apache ZooKeeper 1.0.0 to 3.4.13 and 3.5.0-alpha to 3.5.4-beta. ZooKeeper's getACL() command doesn't check any permission when retrieves the ACLs of the requested node and returns all information contained in the ACL Id field as plaintext string. DigestAuthenticationProvider overloads the Id field with the hash value that is used for user authentication. As a consequence, if Digest Authentication is in use, the unsalted hash value will be disclosed by getACL() request for unauthenticated or unprivileged users. and has a critical [bug](https://issues.apache.org/jira/browse/ZOOKEEPER-2960). <!-- Describe your patch: what did you change in code? How did you fix the problem? --> Zookeeper version is upgraded to 3.4.14 <!-- If there are several relatively logically separate changes in this PR, create a mini-

section for each of them. For example: --> <!-- In each section, please describe design decisions made, including: - Choice of algorithms - Behavioral aspects. What configuration values are acceptable? How are corner cases and error conditions handled, such as when there are insufficient resources? - Class organization and design (how the logic is split between classes, inheritance, composition, design patterns) - Method organization and design (how the logic is split between methods, parameters and return types) - Naming (class, method, API, configuration, HTTP endpoint, names of emitted metrics) --> <!-- It's good to describe an alternative design (or mention an alternative name) for every design (or naming) decision point and compare the alternatives with the designs that you've implemented (or the names you've chosen) to highlight the advantages of the chosen designs and names. --> <!-- If there was a discussion of the design of the feature implemented in this PR elsewhere (e. g. a "Proposal" issue, any other issue, or a thread in the development mailing list), link to that discussion from this PR description and explain what have changed in your final design compared to your original proposal or the consensus version in the end of the discussion. If something hasn't changed since the original discussion, you can omit a detailed discussion of those aspects of the design here, perhaps apart from brief mentioning for the sake of readability of this PR description. --> <!-- Some of the aspects mentioned above may be omitted for simple and small changes. --> <hr> This PR has: - [X] been self-reviewed.<!-- - [ ] using the [concurrency checklist](https://github.com/apache/incubator-druid/blob/master/dev/code-review/concurrency.md) (Remove this item if the PR doesn't have any relation to concurrency.) --> - [ ] added documentation for new or modified features or behaviors. - [ ] added Javadocs for most classes and all non-trivial methods. Linked related entities via Javadoc links. - [ ] added comments explaining the "why" and the intent of the code wherever would not be obvious for an unfamiliar reader. - [ ] added unit tests or modified existing tests to cover new code paths. - [ ] added integration tests. - [ ] been tested in a test Druid cluster. <!-- Check the items by putting "x" in the brackets for the done things. Not all of these items apply to every PR. Remove the items which are not done or not relevant to the PR. None of the items from the checklist above are strictly necessary, but it would be very helpful if you at least self-review the PR. --> <hr>

3. **title:** Zookeeper version is updated.
   body: <!-- Thanks for trying to help us make Apache Druid (incubating) be the best it can be! Please fill out as much of the following information as is possible (where relevant, and remove it when irrelevant) to help make the intention and scope of this PR clear in order to ease review. --> Fixes #8362. <!-- Replace XXXX with the id of the issue fixed in this PR. Remove this section if there is no corresponding issue. Don't reference the issue in the title of this pull-request. --> <!-- If you are a committer, follow the PR action item checklist for committers: https://github.com/apache/incubator-druid/blob/master/dev/committer-instructions.md#pr-and-issue-action-item-checklist-for-committers. --> ### Description <!-- Describe the goal of this PR, what problem are you fixing. If there is a corresponding issue (referenced above), it's not necessary to repeat the description here, however, you may choose to keep one summary sentence. --> Zookeeper version should be upgraded to version 3.4.14 or later due to it has a [vulnerability](https://nvd.nist.gov/vuln/detail/CVE-2019-0201): > An issue is present in Apache ZooKeeper 1.0.0 to 3.4.13 and 3.5.0-alpha to 3.5.4-beta. ZooKeeper's getACL() command doesn't check any permission when retrieves the ACLs of the requested node and returns all information contained in the ACL Id field as plaintext string. DigestAuthenticationProvider overloads the Id field with the hash value that is used for user authentication. As a consequence, if Digest Authentication is in use, the unsalted hash value will be disclosed by getACL() request for unauthenticated or unprivileged users. and has a critical [bug](https://issues.apache.org/jira/browse/ZOOKEEPER-2960). <!-- Describe your patch: what did you change in code? How did you fix the problem? --> Zookeeper version is upgraded to 3.4.14 <!-- If there are several relatively logically separate changes in this PR, create a mini-section for each of them. For example: --> <!-- In each section, please describe design decisions made, including: - Choice of algorithms - Behavioral aspects. What configuration values are acceptable? How are corner cases and error conditions handled, such as when there are insufficient resources? - Class organization and design (how the logic is split between classes, inheritance, composition, design patterns) - Method organization and design (how the logic is split between methods, parameters and return types) - Naming (class, method, API, configuration, HTTP endpoint, names of emitted metrics) --> <!-- It's good to describe an alternative design (or mention an alternative name) for every design (or naming) decision point and compare the alternatives with the designs that you've implemented (or the names you've chosen) to highlight the advantages of the chosen designs and names. --> <!-- If there was a discussion of the design of the feature implemented in this PR elsewhere (e. g. a "Proposal" issue, any other issue, or a thread in the development mailing list), link to that discussion from this PR description and explain what have changed in your final design compared to your original proposal or the consensus version in the end of the discussion. If something hasn't changed since the original discussion, you can omit a detailed discussion of those aspects of the design here, perhaps apart from brief mentioning for the sake of readability of this PR description. --> <!-- Some of the aspects mentioned above may be omitted for simple and small changes. --> <hr> This PR has: - [X] been self-reviewed.<!-- - [ ] using the [concurrency checklist](https://github.com/apache/incubator-druid/blob/master/dev/code-review/concurrency.md) (Remove this item if the PR doesn't have any relation to concurrency.) --> - [ ] added documentation for new or modified features or behaviors. - [ ] added Javadocs for most classes and all non-trivial methods. Linked related entities via Javadoc links. - [ ] added comments explaining the "why" and the intent of the code wherever would not be obvious for an unfamiliar reader. - [ ] added unit tests or modified existing tests to cover new code paths. - [ ] added integration tests. - [ ] been tested in a test Druid cluster. <!-- Check the items by putting "x" in the brackets for the done things. Not all of these items apply to every PR. Remove the items which are not done or not relevant to the PR. None of the items from the checklist above are strictly necessary, but it would be very helpful if you at least self-review the PR. --> <hr>
4. **title:** Zookeeper version is updated.
   body: <!-- Thanks for trying to help us make Apache Druid (incubating) be the best it can be! Please fill out as much of the following information as is possible (where relevant, and remove it when irrelevant) to help make the intention and scope of this PR clear in order to ease review. --> Fixes #8362. <!-- Replace XXXX with the id of the issue fixed in this PR. Remove this section if there is no corresponding issue. Don't reference the issue in the title of this pull-request. --> <!-- If you are a committer, follow the PR action item checklist for committers: https://github.com/apache/incubator-druid/blob/master/dev/committer-instructions.md#pr-and-issue-action-item-checklist-for-committers. --> ### Description <!-- Describe the goal of this PR, what problem are you fixing. If there is a corresponding issue (referenced above), it's not necessary to repeat the description here, however, you may choose to keep one summary sentence. --> Zookeeper version should be upgraded to version 3.4.14 or later due to it has a [vulnerability](https://nvd.nist.gov/vuln/detail/CVE-2019-0201): > An issue is present in Apache ZooKeeper 1.0.0 to 3.4.13 and 3.5.0-alpha to 3.5.4-beta. ZooKeeper's getACL() command doesn't check any permission when retrieves the ACLs of the requested node and returns all information contained in the ACL Id field as plaintext string. DigestAuthenticationProvider overloads the Id field with the hash value that is used for user authentication. As a consequence, if Digest Authentication is in use, the unsalted hash value will be disclosed by getACL() request for unauthenticated or unprivileged users. and has a critical [bug](https://issues.apache.org/jira/browse/ZOOKEEPER-2960). <!-- Describe your patch: what did you change in code? How did you fix the problem? --> Zookeeper version is upgraded to 3.4.14 <!-- If there are several relatively logically separate changes in this PR, create a mini-section for each of them. For example: --> <!-- In each section, please describe design decisions made, including: - Choice of algorithms - Behavioral aspects. What configuration values are acceptable? How are corner cases and error conditions handled, such as when there are insufficient resources? - Class organization and design (how the logic is split between classes, inheritance, composition, design patterns) - Method organization and design (how the logic is split between methods, parameters and return types) - Naming (class, method, API, configuration, HTTP endpoint, names of emitted metrics) --> <!-- It's good to describe an alternative design (or mention an alternative name) for every design (or naming) decision point and compare the alternatives with the designs that you've implemented (or the names you've chosen) to highlight the advantages of the chosen designs and names. --> <!-- If there was a discussion of the design of the feature implemented in this PR elsewhere (e. g. a "Proposal" issue, any other issue, or a thread in the development mailing list), link to that discussion from this PR description and explain what have changed in your final design compared to your original proposal or the consensus version in the end of the discussion. If something hasn't changed since the original discussion, you can omit a detailed discussion of those aspects of the design here, perhaps apart from brief mentioning for the sake of readability of this PR description. --> <!-- Some of the aspects mentioned above may be omitted for simple and

small changes. --> <hr> This PR has: - [X] been self-reviewed.<!-- - [ ] using the [concurrency checklist](https://github.com/apache/incubator-druid/blob/master/dev/code-review/concurrency.md) (Remove this item if the PR doesn't have any relation to concurrency.) --> - [ ] added documentation for new or modified features or behaviors. - [ ] added Javadocs for most classes and all non-trivial methods. Linked related entities via Javadoc links. - [ ] added comments explaining the "why" and the intent of the code wherever would not be obvious for an unfamiliar reader. - [ ] added unit tests or modified existing tests to cover new code paths. - [ ] added integration tests. - [ ] been tested in a test Druid cluster. <!-- Check the items by putting "x" in the brackets for the done things. Not all of these items apply to every PR. Remove the items which are not done or not relevant to the PR. None of the items from the checklist above are strictly necessary, but it would be very helpful if you at least self-review the PR. --> <hr>

5. **title:** Zookeeper version is updated.
   body: <!-- Thanks for trying to help us make Apache Druid (incubating) be the best it can be! Please fill out as much of the following information as is possible (where relevant, and remove it when irrelevant) to help make the intention and scope of this PR clear in order to ease review. --> Fixes #8362. <!-- Replace XXXX with the id of the issue fixed in this PR. Remove this section if there is no corresponding issue. Don't reference the issue in the title of this pull-request. --> <!-- If you are a committer, follow the PR action item checklist for committers: https://github.com/apache/incubator-druid/blob/master/dev/committer-instructions.md#pr-and-issue-action-item-checklist-for-committers. --> ### Description <!-- Describe the goal of this PR, what problem are you fixing. If there is a corresponding issue (referenced above), it's not necessary to repeat the description here, however, you may choose to keep one summary sentence. --> Zookeeper version should be upgraded to version 3.4.14 or later due to it has a [vulnerability](https://nvd.nist.gov/vuln/detail/CVE-2019-0201): > An issue is present in Apache ZooKeeper 1.0.0 to 3.4.13 and 3.5.0-alpha to 3.5.4-beta. ZooKeeper's getACL() command doesn't check any permission when retrieves the ACLs of the requested node and returns all information contained in the ACL Id field as plaintext string. DigestAuthenticationProvider overloads the Id field with the hash value that is used for user authentication. As a consequence, if Digest Authentication is in use, the unsalted hash value will be disclosed by getACL() request for unauthenticated or unprivileged users. and has a critical [bug](https://issues.apache.org/jira/browse/ZOOKEEPER-2960). <!-- Describe your patch: what did you change in code? How did you fix the problem? --> Zookeeper version is upgraded to 3.4.14 <!-- If there are several relatively logically separate changes in this PR, create a mini-section for each of them. For example: --> <!-- In each section, please describe design decisions made, including: - Choice of algorithms - Behavioral aspects. What configuration values are acceptable? How are corner cases and error conditions handled, such as when there are insufficient resources? - Class organization and design (how the logic is split between classes, inheritance, composition, design patterns) - Method organization and design (how the logic is split between methods, parameters and return types) - Naming (class, method, API, configuration, HTTP endpoint, names of emitted metrics) --> <!-- It's good to describe an alternative design (or mention an alternative name) for every design (or naming) decision point and compare the alternatives with the designs that you've implemented (or the names you've chosen) to highlight the advantages of the chosen designs and names. --> <!-- If there was a discussion of the design of the feature implemented in this PR elsewhere (e. g. a "Proposal" issue, any other issue, or a thread in the development mailing list), link to that discussion from this PR description and explain what have changed in your final design compared to your original proposal or the consensus version in the end of the discussion. If something hasn't changed since the original discussion, you can omit a detailed discussion of those aspects of the design here, perhaps apart from brief mentioning for the sake of readability of this PR description. --> <!-- Some of the aspects mentioned above may be omitted for simple and small changes. --> <hr> This PR has: - [X] been self-reviewed.<!-- - [ ] using the [concurrency checklist](https://github.com/apache/incubator-druid/blob/master/dev/code-review/concurrency.md) (Remove this item if the PR doesn't have any relation to concurrency.) --> - [ ] added documentation for new or modified features or behaviors. - [ ] added Javadocs for most classes and all non-trivial methods. Linked related entities via Javadoc links. - [ ] added comments explaining the "why" and the intent of the code wherever would not be obvious for an unfamiliar reader. - [ ] added unit tests or modified existing tests to cover new code paths. - [ ] added integration tests. - [ ] been tested in a test Druid cluster. <!-- Check the items by putting "x" in the brackets for the done things. Not all of these items apply to every PR. Remove the items which are not done or not relevant to the PR. None of the items from the checklist above are strictly necessary, but it would be very helpful if you at least self-review the PR. --> <hr>
6. **title:** Zookeeper version is updated.
   body: <!-- Thanks for trying to help us make Apache Druid (incubating) be the best it can be! Please fill out as much of the following information as is possible (where relevant, and remove it when irrelevant) to help make the intention and scope of this PR clear in order to ease review. --> Fixes #8362. <!-- Replace XXXX with the id of the issue fixed in this PR. Remove this section if there is no corresponding issue. Don't reference the issue in the title of this pull-request. --> <!-- If you are a committer, follow the PR action item checklist for committers: https://github.com/apache/incubator-druid/blob/master/dev/committer-instructions.md#pr-and-issue-action-item-checklist-for-committers. --> ### Description <!-- Describe the goal of this PR, what problem are you fixing. If there is a corresponding issue (referenced above), it's not necessary to repeat the description here, however, you may choose to keep one summary sentence. --> Zookeeper version should be upgraded to version 3.4.14 or later due to it has a [vulnerability](https://nvd.nist.gov/vuln/detail/CVE-2019-0201): > An issue is present in Apache ZooKeeper 1.0.0 to 3.4.13 and 3.5.0-alpha to 3.5.4-beta. ZooKeeper's getACL() command doesn't check any permission when retrieves the ACLs of the requested node and returns all information contained in the ACL Id field as plaintext string. DigestAuthenticationProvider overloads the Id field with the hash value that is used for user authentication. As a consequence, if Digest Authentication is in use, the unsalted hash value will be disclosed by getACL() request for unauthenticated or unprivileged users. and has a critical [bug](https://issues.apache.org/jira/browse/ZOOKEEPER-2960). <!-- Describe your patch: what did you change in code? How did you fix the problem? --> Zookeeper version is upgraded to 3.4.14 <!-- If there are several relatively logically separate changes in this PR, create a mini-section for each of them. For example: --> <!-- In each section, please describe design decisions made, including: - Choice of algorithms - Behavioral aspects. What configuration values are acceptable? How are corner cases and error conditions handled, such as when there are insufficient resources? - Class organization and design (how the logic is split between classes, inheritance, composition, design patterns) - Method organization and design (how the logic is split between methods, parameters and return types) - Naming (class, method, API, configuration, HTTP endpoint, names of emitted metrics) --> <!-- It's good to describe an alternative design (or mention an alternative name) for every design (or naming) decision point and compare the alternatives with the designs that you've implemented (or the names you've chosen) to highlight the advantages of the chosen designs and names. --> <!-- If there was a discussion of the design of the feature implemented in this PR elsewhere (e. g. a "Proposal" issue, any other issue, or a thread in the development mailing list), link to that discussion from this PR description and explain what have changed in your final design compared to your original proposal or the consensus version in the end of the discussion. If something hasn't changed since the original discussion, you can omit a detailed discussion of those aspects of the design here, perhaps apart from brief mentioning for the sake of readability of this PR description. --> <!-- Some of the aspects mentioned above may be omitted for simple and small changes. --> <hr> This PR has: - [X] been self-reviewed.<!-- - [ ] using the [concurrency checklist](https://github.com/apache/incubator-druid/blob/master/dev/code-review/concurrency.md) (Remove this item if the PR doesn't have any relation to concurrency.) --> - [ ] added documentation for new or modified features or behaviors. - [ ] added Javadocs for most classes and all non-trivial methods. Linked related entities via Javadoc links. - [ ] added comments explaining the "why" and the intent of the code wherever would not be obvious for an unfamiliar reader. - [ ] added unit tests or modified existing tests to cover new code paths. - [ ] added integration tests. - [ ] been tested in a test Druid cluster. <!-- Check the items by putting "x" in the brackets for the done things. Not all of these items apply to every PR. Remove the items which are not done or not relevant to the PR. None of the items from the checklist above are strictly necessary, but it would be very helpful if you at least self-review the PR. --> <hr>
7. **title:** Zookeeper version is updated.
   body: <!-- Thanks for trying to help us make Apache Druid (incubating) be the best it can be! Please fill out as much of the following information as is possible (where relevant, and remove it when irrelevant) to help make the intention and scope of this PR clear in order to ease

review. --> Fixes #8362. <!-- Replace XXXX with the id of the issue fixed in this PR. Remove this section if there is no corresponding issue. Don't reference the issue in the title of this pull-request. --> <!-- If you are a committer, follow the PR action item checklist for committers: https://github.com/apache/incubator-druid/blob/master/dev/committer-instructions.md#pr-and-issue-action-item-checklist-for-committers. --> ### Description <!-- Describe the goal of this PR, what problem are you fixing. If there is a corresponding issue (referenced above), it's not necessary to repeat the description here, however, you may choose to keep one summary sentence. --> Zookeeper version should be upgraded to version 3.4.14 or later due to it has a [vulnerability](https://nvd.nist.gov/vuln/detail/CVE-2019-0201): > An issue is present in Apache ZooKeeper 1.0.0 to 3.4.13 and 3.5.0-alpha to 3.5.4-beta. ZooKeeper's getACL() command doesn't check any permission when retrieves the ACLs of the requested node and returns all information contained in the ACL Id field as plaintext string. DigestAuthenticationProvider overloads the Id field with the hash value that is used for user authentication. As a consequence, if Digest Authentication is in use, the unsalted hash value will be disclosed by getACL() request for unauthenticated or unprivileged users. and has a critical [bug] (https://issues.apache.org/jira/browse/ZOOKEEPER-2960). <!-- Describe your patch: what did you change in code? How did you fix the problem? --> Zookeeper version is upgraded to 3.4.14 <!-- If there are several relatively logically separate changes in this PR, create a mini-section for each of them. For example: --> <!-- In each section, please describe design decisions made, including: - Choice of algorithms - Behavioral aspects. What configuration values are acceptable? How are corner cases and error conditions handled, such as when there are insufficient resources? - Class organization and design (how the logic is split between classes, inheritance, composition, design patterns) - Method organization and design (how the logic is split between methods, parameters and return types) - Naming (class, method, API, configuration, HTTP endpoint, names of emitted metrics) --> <!-- It's good to describe an alternative design (or mention an alternative name) for every design (or naming) decision point and compare the alternatives with the designs that you've implemented (or the names you've chosen) to highlight the advantages of the chosen designs and names. --> <!-- If there was a discussion of the design of the feature implemented in this PR elsewhere (e. g. a "Proposal" issue, any other issue, or a thread in the development mailing list), link to that discussion from this PR description and explain what have changed in your final design compared to your original proposal or the consensus version in the end of the discussion. If something hasn't changed since the original discussion, you can omit a detailed discussion of those aspects of the design here, perhaps apart from brief mentioning for the sake of readability of this PR description. --> <!-- Some of the aspects mentioned above may be omitted for simple and small changes. --> <hr> This PR has: - [X] been self-reviewed.<!-- - [ ] using the [concurrency checklist](https://github.com/apache/incubator-druid/blob/master/dev/code-review/concurrency.md) (Remove this item if the PR doesn't have any relation to concurrency.) --> - [ ] added documentation for new or modified features or behaviors. - [ ] added Javadocs for most classes and all non-trivial methods. Linked related entities via Javadoc links. - [ ] added comments explaining the "why" and the intent of the code wherever would not be obvious for an unfamiliar reader. - [ ] added unit tests or modified existing tests to cover new code paths. - [ ] added integration tests. - [ ] been tested in a test Druid cluster. <!-- Check the items by putting "x" in the brackets for the done things. Not all of these items apply to every PR. Remove the items which are not done or not relevant to the PR. None of the items from the checklist above are strictly necessary, but it would be very helpful if you at least self-review the PR. --> <hr>

8. **title:** Zookeeper version is updated.
   body: <!-- Thanks for trying to help us make Apache Druid (incubating) be the best it can be! Please fill out as much of the following information as is possible (where relevant, and remove it when irrelevant) to help make the intention and scope of this PR clear in order to ease review. --> Fixes #8362. <!-- Replace XXXX with the id of the issue fixed in this PR. Remove this section if there is no corresponding issue. Don't reference the issue in the title of this pull-request. --> <!-- If you are a committer, follow the PR action item checklist for committers: https://github.com/apache/incubator-druid/blob/master/dev/committer-instructions.md#pr-and-issue-action-item-checklist-for-committers. --> ### Description <!-- Describe the goal of this PR, what problem are you fixing. If there is a corresponding issue (referenced above), it's not necessary to repeat the description here, however, you may choose to keep one summary sentence. --> Zookeeper version should be upgraded to version 3.4.14 or later due to it has a [vulnerability](https://nvd.nist.gov/vuln/detail/CVE-2019-0201): > An issue is present in Apache ZooKeeper 1.0.0 to 3.4.13 and 3.5.0-alpha to 3.5.4-beta. ZooKeeper's getACL() command doesn't check any permission when retrieves the ACLs of the requested node and returns all information contained in the ACL Id field as plaintext string. DigestAuthenticationProvider overloads the Id field with the hash value that is used for user authentication. As a consequence, if Digest Authentication is in use, the unsalted hash value will be disclosed by getACL() request for unauthenticated or unprivileged users. and has a critical [bug] (https://issues.apache.org/jira/browse/ZOOKEEPER-2960). <!-- Describe your patch: what did you change in code? How did you fix the problem? --> Zookeeper version is upgraded to 3.4.14 <!-- If there are several relatively logically separate changes in this PR, create a mini-section for each of them. For example: --> <!-- In each section, please describe design decisions made, including: - Choice of algorithms - Behavioral aspects. What configuration values are acceptable? How are corner cases and error conditions handled, such as when there are insufficient resources? - Class organization and design (how the logic is split between classes, inheritance, composition, design patterns) - Method organization and design (how the logic is split between methods, parameters and return types) - Naming (class, method, API, configuration, HTTP endpoint, names of emitted metrics) --> <!-- It's good to describe an alternative design (or mention an alternative name) for every design (or naming) decision point and compare the alternatives with the designs that you've implemented (or the names you've chosen) to highlight the advantages of the chosen designs and names. --> <!-- If there was a discussion of the design of the feature implemented in this PR elsewhere (e. g. a "Proposal" issue, any other issue, or a thread in the development mailing list), link to that discussion from this PR description and explain what have changed in your final design compared to your original proposal or the consensus version in the end of the discussion. If something hasn't changed since the original discussion, you can omit a detailed discussion of those aspects of the design here, perhaps apart from brief mentioning for the sake of readability of this PR description. --> <!-- Some of the aspects mentioned above may be omitted for simple and small changes. --> <hr> This PR has: - [X] been self-reviewed.<!-- - [ ] using the [concurrency checklist](https://github.com/apache/incubator-druid/blob/master/dev/code-review/concurrency.md) (Remove this item if the PR doesn't have any relation to concurrency.) --> - [ ] added documentation for new or modified features or behaviors. - [ ] added Javadocs for most classes and all non-trivial methods. Linked related entities via Javadoc links. - [ ] added comments explaining the "why" and the intent of the code wherever would not be obvious for an unfamiliar reader. - [ ] added unit tests or modified existing tests to cover new code paths. - [ ] added integration tests. - [ ] been tested in a test Druid cluster. <!-- Check the items by putting "x" in the brackets for the done things. Not all of these items apply to every PR. Remove the items which are not done or not relevant to the PR. None of the items from the checklist above are strictly necessary, but it would be very helpful if you at least self-review the PR. --> <hr>

github_pulls_comments:

1. This is a somewhat duplicate of #8177, but maybe still worth going ahead and doing since the other is tied up in the curator issue. After #7998 and #8306 (which was just merged and the cause of your conflicts), `LICENSE.BINARY` and `NOTICE.BINARY` are both autogenerated from the contents of `licenses.yaml` at the root of the project. #8367 will add instructions to the PR template indicating that this should be done if dependency versions are bumped, but it is not merged yet. Could you please fix conflicts update the zookeeper entry 'version' and 'notice' text in [licenses.yaml](https://github.com/apache/incubator-druid/blob/master/licenses.yaml#L1453)? For the notice, it looks like changing to: ``` Apache ZooKeeper Copyright 2009-2019 The Apache Software Foundation ``` [should be sufficient] (https://github.com/apache/zookeeper/blob/4c25d480e66aadd371de8bd2fd8da255ac140bcf/NOTICE.txt).
2. @clintropolis I've updated my PR.
3. I've updated my PR.

github_pulls_reviews:

1. Please update the `notices` section with the copyright information I mentioned in previous comment: ``` Apache ZooKeeper Copyright 2009-2019 The Apache Software Foundation ```
2. Oops, you've updated the notice for the zookeeper bundled in the hadoop-client, instead of the one in core druid that you actually updated the version for (notice starts on line 1458).
3. 👍

jira_issues:

1. **summary:** The dataDir and dataLogDir are used opposingly
   description: _emphasized text_After upgrading from zookeeper 3.4.5, to 3.4.11, without editing {{zoo.cfg}}, the new version of the server tries to use the {{dataDir}} as the {{dataLogDir}}, and the {{dataLogDir}} as the {{dataDir}}. Or at least some parts of the server. Configuration file has: {noformat} $ grep -i data /etc/zookeeper/zoo.cfg dataLogDir=/var/lib/zookeeper/datalog dataDir=/var/lib/zookeeper/data {noformat} But runtime configuration has: {noformat} $ echo conf | nc localhost 2181 | grep -i data dataDir=/var/lib/zookeeper/datalog/version-2 dataLogDir=/var/lib/zookeeper/data/version-2 {noformat} Also, I got this in the debug logs, so clearly some parts of the server confuse things. {noformat} [PurgeTask:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/datalog snapDir:/var/lib/zookeeper/data [main:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/data snapDir:/var/lib/zookeeper/datalog {noformat} I tried to look in the code for wrong uses of the directories. I only found [ZookeeperServer.java|https://github.com/apache/zookeeper/blob/master/src/java/main/org/apache/zookeeper/server/ZooKeeperServer.java#L227] is passing the arguments to {{FileTxnSnapLog}} in the wrong order, but the code comment says that this is legacy only for tests, so I assume it isn't the cause for my case.
2. **summary:** The dataDir and dataLogDir are used opposingly
   description: _emphasized text_After upgrading from zookeeper 3.4.5, to 3.4.11, without editing {{zoo.cfg}}, the new version of the server tries to use the {{dataDir}} as the {{dataLogDir}}, and the {{dataLogDir}} as the {{dataDir}}. Or at least some parts of the server. Configuration file has: {noformat} $ grep -i data /etc/zookeeper/zoo.cfg dataLogDir=/var/lib/zookeeper/datalog dataDir=/var/lib/zookeeper/data {noformat} But runtime configuration has: {noformat} $ echo conf | nc localhost 2181 | grep -i data dataDir=/var/lib/zookeeper/datalog/version-2 dataLogDir=/var/lib/zookeeper/data/version-2 {noformat} Also, I got this in the debug logs, so clearly some parts of the server confuse things. {noformat} [PurgeTask:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/datalog snapDir:/var/lib/zookeeper/data [main:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/data snapDir:/var/lib/zookeeper/datalog {noformat} I tried to look in the code for wrong uses of the directories. I only found [ZookeeperServer.java|https://github.com/apache/zookeeper/blob/master/src/java/main/org/apache/zookeeper/server/ZooKeeperServer.java#L227] is passing the arguments to {{FileTxnSnapLog}} in the wrong order, but the code comment says that this is legacy only for tests, so I assume it isn't the cause for my case.
3. **summary:** The dataDir and dataLogDir are used opposingly
   description: _emphasized text_After upgrading from zookeeper 3.4.5, to 3.4.11, without editing {{zoo.cfg}}, the new version of the server tries to use the {{dataDir}} as the {{dataLogDir}}, and the {{dataLogDir}} as the {{dataDir}}. Or at least some parts of the server. Configuration file has: {noformat} $ grep -i data /etc/zookeeper/zoo.cfg dataLogDir=/var/lib/zookeeper/datalog dataDir=/var/lib/zookeeper/data {noformat} But runtime configuration has: {noformat} $ echo conf | nc localhost 2181 | grep -i data dataDir=/var/lib/zookeeper/datalog/version-2 dataLogDir=/var/lib/zookeeper/data/version-2 {noformat} Also, I got this in the debug logs, so clearly some parts of the server confuse things. {noformat} [PurgeTask:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/datalog snapDir:/var/lib/zookeeper/data [main:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/data snapDir:/var/lib/zookeeper/datalog {noformat} I tried to look in the code for wrong uses of the directories. I only found [ZookeeperServer.java|https://github.com/apache/zookeeper/blob/master/src/java/main/org/apache/zookeeper/server/ZooKeeperServer.java#L227] is passing the arguments to {{FileTxnSnapLog}} in the wrong order, but the code comment says that this is legacy only for tests, so I assume it isn't the cause for my case.
   label: code-design
4. **summary:** The dataDir and dataLogDir are used opposingly
   description: _emphasized text_After upgrading from zookeeper 3.4.5, to 3.4.11, without editing {{zoo.cfg}}, the new version of the server tries to use the {{dataDir}} as the {{dataLogDir}}, and the {{dataLogDir}} as the {{dataDir}}. Or at least some parts of the server. Configuration file has: {noformat} $ grep -i data /etc/zookeeper/zoo.cfg dataLogDir=/var/lib/zookeeper/datalog dataDir=/var/lib/zookeeper/data {noformat} But runtime configuration has: {noformat} $ echo conf | nc localhost 2181 | grep -i data dataDir=/var/lib/zookeeper/datalog/version-2 dataLogDir=/var/lib/zookeeper/data/version-2 {noformat} Also, I got this in the debug logs, so clearly some parts of the server confuse things. {noformat} [PurgeTask:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/datalog snapDir:/var/lib/zookeeper/data [main:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/data snapDir:/var/lib/zookeeper/datalog {noformat} I tried to look in the code for wrong uses of the directories. I only found [ZookeeperServer.java|https://github.com/apache/zookeeper/blob/master/src/java/main/org/apache/zookeeper/server/ZooKeeperServer.java#L227] is passing the arguments to {{FileTxnSnapLog}} in the wrong order, but the code comment says that this is legacy only for tests, so I assume it isn't the cause for my case.
5. **summary:** The dataDir and dataLogDir are used opposingly
   description: _emphasized text_After upgrading from zookeeper 3.4.5, to 3.4.11, without editing {{zoo.cfg}}, the new version of the server tries to use the {{dataDir}} as the {{dataLogDir}}, and the {{dataLogDir}} as the {{dataDir}}. Or at least some parts of the server. Configuration file has: {noformat} $ grep -i data /etc/zookeeper/zoo.cfg dataLogDir=/var/lib/zookeeper/datalog dataDir=/var/lib/zookeeper/data {noformat} But runtime configuration has: {noformat} $ echo conf | nc localhost 2181 | grep -i data dataDir=/var/lib/zookeeper/datalog/version-2 dataLogDir=/var/lib/zookeeper/data/version-2 {noformat} Also, I got this in the debug logs, so clearly some parts of the server confuse things. {noformat} [PurgeTask:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/datalog snapDir:/var/lib/zookeeper/data [main:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/data snapDir:/var/lib/zookeeper/datalog {noformat} I tried to look in the code for wrong uses of the directories. I only found [ZookeeperServer.java|https://github.com/apache/zookeeper/blob/master/src/java/main/org/apache/zookeeper/server/ZooKeeperServer.java#L227] is passing the arguments to {{FileTxnSnapLog}} in the wrong order, but the code comment says that this is legacy only for tests, so I assume it isn't the cause for my case.
6. **summary:** The dataDir and dataLogDir are used opposingly
   description: _emphasized text_After upgrading from zookeeper 3.4.5, to 3.4.11, without editing {{zoo.cfg}}, the new version of the server tries to use the {{dataDir}} as the {{dataLogDir}}, and the {{dataLogDir}} as the {{dataDir}}. Or at least some parts of the server. Configuration file has: {noformat} $ grep -i data /etc/zookeeper/zoo.cfg dataLogDir=/var/lib/zookeeper/datalog dataDir=/var/lib/zookeeper/data {noformat} But runtime configuration has: {noformat} $ echo conf | nc localhost 2181 | grep -i data dataDir=/var/lib/zookeeper/datalog/version-2 dataLogDir=/var/lib/zookeeper/data/version-2 {noformat} Also, I got this in the debug logs, so clearly some parts of the server confuse things. {noformat} [PurgeTask:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/datalog snapDir:/var/lib/zookeeper/data [main:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/data snapDir:/var/lib/zookeeper/datalog {noformat} I tried to look in the code for wrong uses of the directories. I only found [ZookeeperServer.java|https://github.com/apache/zookeeper/blob/master/src/java/main/org/apache/zookeeper/server/ZooKeeperServer.java#L227]

is passing the arguments to {{FileTxnSnapLog}} in the wrong order, but the code comment says that this is legacy only for tests, so I assume it isn't the cause for my case.

7. **summary:** The dataDir and dataLogDir are used opposingly
   description: _emphasized text_After upgrading from zookeeper 3.4.5, to 3.4.11, without editing {{zoo.cfg}}, the new version of the server tries to use the {{dataDir}} as the {{dataLogDir}}, and the {{dataLogDir}} as the {{dataDir}}. Or at least some parts of the server. Configuration file has: {noformat} $ grep -i data /etc/zookeeper/zoo.cfg dataLogDir=/var/lib/zookeeper/datalog dataDir=/var/lib/zookeeper/data {noformat} But runtime configuration has: {noformat} $ echo conf | nc localhost 2181 | grep -i data dataDir=/var/lib/zookeeper/datalog/version-2 dataLogDir=/var/lib/zookeeper/data/version-2 {noformat} Also, I got this in the debug logs, so clearly some parts of the server confuse things. {noformat} [PurgeTask:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/datalog snapDir:/var/lib/zookeeper/data [main:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/data snapDir:/var/lib/zookeeper/datalog {noformat} I tried to look in the code for wrong uses of the directories. I only found [ZookeeperServer.java|https://github.com/apache/zookeeper/blob/master/src/java/main/org/apache/zookeeper/server/ZooKeeperServer.java#L227] is passing the arguments to {{FileTxnSnapLog}} in the wrong order, but the code comment says that this is legacy only for tests, so I assume it isn't the cause for my case.
8. **summary:** The dataDir and dataLogDir are used opposingly
   description: _emphasized text_After upgrading from zookeeper 3.4.5, to 3.4.11, without editing {{zoo.cfg}}, the new version of the server tries to use the {{dataDir}} as the {{dataLogDir}}, and the {{dataLogDir}} as the {{dataDir}}. Or at least some parts of the server. Configuration file has: {noformat} $ grep -i data /etc/zookeeper/zoo.cfg dataLogDir=/var/lib/zookeeper/datalog dataDir=/var/lib/zookeeper/data {noformat} But runtime configuration has: {noformat} $ echo conf | nc localhost 2181 | grep -i data dataDir=/var/lib/zookeeper/datalog/version-2 dataLogDir=/var/lib/zookeeper/data/version-2 {noformat} Also, I got this in the debug logs, so clearly some parts of the server confuse things. {noformat} [PurgeTask:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/datalog snapDir:/var/lib/zookeeper/data [main:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/data snapDir:/var/lib/zookeeper/datalog {noformat} I tried to look in the code for wrong uses of the directories. I only found [ZookeeperServer.java|https://github.com/apache/zookeeper/blob/master/src/java/main/org/apache/zookeeper/server/ZooKeeperServer.java#L227] is passing the arguments to {{FileTxnSnapLog}} in the wrong order, but the code comment says that this is legacy only for tests, so I assume it isn't the cause for my case.
9. **summary:** The dataDir and dataLogDir are used opposingly
   description: _emphasized text_After upgrading from zookeeper 3.4.5, to 3.4.11, without editing {{zoo.cfg}}, the new version of the server tries to use the {{dataDir}} as the {{dataLogDir}}, and the {{dataLogDir}} as the {{dataDir}}. Or at least some parts of the server. Configuration file has: {noformat} $ grep -i data /etc/zookeeper/zoo.cfg dataLogDir=/var/lib/zookeeper/datalog dataDir=/var/lib/zookeeper/data {noformat} But runtime configuration has: {noformat} $ echo conf | nc localhost 2181 | grep -i data dataDir=/var/lib/zookeeper/datalog/version-2 dataLogDir=/var/lib/zookeeper/data/version-2 {noformat} Also, I got this in the debug logs, so clearly some parts of the server confuse things. {noformat} [PurgeTask:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/datalog snapDir:/var/lib/zookeeper/data [main:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/data snapDir:/var/lib/zookeeper/datalog {noformat} I tried to look in the code for wrong uses of the directories. I only found [ZookeeperServer.java|https://github.com/apache/zookeeper/blob/master/src/java/main/org/apache/zookeeper/server/ZooKeeperServer.java#L227] is passing the arguments to {{FileTxnSnapLog}} in the wrong order, but the code comment says that this is legacy only for tests, so I assume it isn't the cause for my case.
10. **summary:** The dataDir and dataLogDir are used opposingly
   description: _emphasized text_After upgrading from zookeeper 3.4.5, to 3.4.11, without editing {{zoo.cfg}}, the new version of the server tries to use the {{dataDir}} as the {{dataLogDir}}, and the {{dataLogDir}} as the {{dataDir}}. Or at least some parts of the server. Configuration file has: {noformat} $ grep -i data /etc/zookeeper/zoo.cfg dataLogDir=/var/lib/zookeeper/datalog dataDir=/var/lib/zookeeper/data {noformat} But runtime configuration has: {noformat} $ echo conf | nc localhost 2181 | grep -i data dataDir=/var/lib/zookeeper/datalog/version-2 dataLogDir=/var/lib/zookeeper/data/version-2 {noformat} Also, I got this in the debug logs, so clearly some parts of the server confuse things. {noformat} [PurgeTask:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/datalog snapDir:/var/lib/zookeeper/data [main:FileTxnSnapLog@79] - Opening datadir:/var/lib/zookeeper/data snapDir:/var/lib/zookeeper/datalog {noformat} I tried to look in the code for wrong uses of the directories. I only found [ZookeeperServer.java|https://github.com/apache/zookeeper/blob/master/src/java/main/org/apache/zookeeper/server/ZooKeeperServer.java#L227] is passing the arguments to {{FileTxnSnapLog}} in the wrong order, but the code comment says that this is legacy only for tests, so I assume it isn't the cause for my case.

jira_issues_comments:

1. Any comments on this? Has anyone been able to confirm it?
2. Hi [~danmilon] I'm taking a look. Andor
3. **body:** Hi [~danmilon] Looks like you're right. QuorumPeerMain mixes the 2 directories and if the server is running in an ensemble, logs are going to be created in dataDir and snapshots are in dataLogDir. What's even more confusing is that it works right in standalone mode. QuorumPeerMain@139 (non-standalone mode): {code:java} runFromConfig() { ... quorumPeer.setTxnFactory(new FileTxnSnapLog(new File(config.getDataDir()), new File(config.getDataLogDir()))); {code} ZookeeperServerMain@112 (standalone mode): {code:java} runFromConfig() { ... txnLog = new FileTxnSnapLog(new File(config.dataLogDir), new File(config.dataDir)); {code} The latter is the right way of doing it: FileTxnSnapLog expects dataLogDir (txn logs) in the 1st and dataDir (snapshots) in the 2nd argument.
   label: code-design
4. Looks like a pretty serious one. This was introduced in branch-3.4 only and released in 3.4.11, because a small refactoring has been made in instantiation of QuorumPeer and now FileTxnSnapLog is created with wrong order of directories. As part of: https://issues.apache.org/jira/browse/ZOOKEEPER-2355 Github pull request: https://github.com/apache/zookeeper/pull/304/files#diff-0b7ebe0e5e2011bb4ba7a9f352f948f7 I believe that users should revert to 3.4.10 and a new release has to be created.
5. GitHub user anmolnar opened a pull request: https://github.com/apache/zookeeper/pull/441 ZOOKEEPER-2960. The dataDir and dataLogDir are used opposingly Previously a small refactoring has swapped the usage of dataDir and dataLogDir when QuorumPeerMain instantiates FileTxnSnapLog class. This PR fixes it + adds unit test for verification. You can merge this pull request into a Git repository by running: $ git pull https://github.com/anmolnar/zookeeper ZOOKEEPER-2960 Alternatively you can review and apply these changes as the patch at: https://github.com/apache/zookeeper/pull/441.patch To close this pull request, make a commit to your master/trunk branch with (at least) the following in the commit message: This closes #441 ---- commit 516147499541f7eb2b30f7a8cb4f50ed204469bb Author: Andor Molnar <andor@...> Date: 2018-01-03T14:47:26Z ZOOKEEPER-2960. Fixed usage of dataDir and dataLogDir when instantiating FileTxnSnapLog ----
6. +1 overall. GitHub Pull Request Build +1 @author. The patch does not contain any @author tags. +1 tests included. The patch appears to include 3 new or modified tests. +1 javadoc. The javadoc tool did not generate any warning messages. +1 javac. The applied patch does not increase the total number of javac compiler warnings. +1 findbugs. The patch does not introduce any new Findbugs (version 3.0.1) warnings. +1 release audit. The applied patch does not increase the total number of release audit warnings. +1 core tests. The patch passed core unit tests. +1 contrib tests. The patch passed contrib unit tests. Test results: https://builds.apache.org/job/PreCommit-ZOOKEEPER-github-pr-build/1395//testReport/ Findbugs warnings: https://builds.apache.org/job/PreCommit-ZOOKEEPER-github-pr-

build/1395//artifact/trunk/build/test/findbugs/newPatchFindbugsWarnings.html Console output: https://builds.apache.org/job/PreCommit-ZOOKEEPER-github-pr-build/1395//console This message is automatically generated.

7. Thanks [~andor] for submitting your analysis and patch for this issue. I also will have a look on this issue very soon.
8. Thanks [~danmilon] for reporting this and [~andor] for submitting a patch. The fix looks good and I gave it a +1 on GitHub. I'm wondering if we need to provide a way for users running 3.4.11 to be backwards compatible or we can just push a new release with this change (and hopefully ZOOKEEPER-2249 as well). Perhaps we should introduce a check of some kind? If datalogdir is different that datadir and snapshots exist in datalogdir we throw an exception and quit. What do you think?
9. Github user skamille commented on the issue: https://github.com/apache/zookeeper/pull/441 +1 gonna attempt to commit this
10. Github user anmolnar closed the pull request at: https://github.com/apache/zookeeper/pull/441
11. Thank you guys :)
12. Github user anmolnar commented on the issue: https://github.com/apache/zookeeper/pull/475 I know what the problem is. 3.4.11 has a bug that swaps dataDir and dataLogDir from config file. Sorry, it's only mentioned briefly on the release page: http://zookeeper.apache.org/releases.html > WARNING: ZOOKEEPER-2960 was recently identified as a regression in 3.4.11 ... Please wait for 3.4.12 for the fix and in the meantime you can probably workaround it with your fix, but it should not be merged in the codebase.
13. ijuma opened a new pull request #4678: MINOR: Revert to ZooKeeper 3.4.10 due to ZOOKEEPER-2960 URL: https://github.com/apache/kafka/pull/4678 It's a critical bug that only affects the server, but we don't have an easy way to use 3.4.11 for the zookeeper client only. For reference the upgrade to ZooKeeper 3.4.11 was done via 2652565d429138c58. Testing strategy: relying on existing tests and reverted a change to a test to pass with 3.4.10. ### Committer Checklist (excluded from commit message) - [ ] Verify design and implementation - [ ] Verify test coverage and CI build status - [ ] Verify documentation (including upgrade notes) -- This is an automated message from the Apache Git Service. To respond to the message, please log on GitHub and use the URL above to go to the specific comment. For queries about this service, please contact Infrastructure at: users@infra.apache.org
14. ijuma closed pull request #4678: MINOR: Revert to ZooKeeper 3.4.10 due to ZOOKEEPER-2960 URL: https://github.com/apache/kafka/pull/4678 This is a PR merged from a forked repository. As GitHub hides the original diff on merge, it is displayed below for the sake of provenance: As this is a foreign pull request (from a fork), the diff is supplied below (as it won't show otherwise due to GitHub magic): diff --git a/core/src/test/scala/unit/kafka/server/ServerShutdownTest.scala b/core/src/test/scala/unit/kafka/server/ServerShutdownTest.scala index 596c3539f83..5e33816783a 100755 --- a/core/src/test/scala/unit/kafka/server/ServerShutdownTest.scala +++ b/core/src/test/scala/unit/kafka/server/ServerShutdownTest.scala @@ -23,6 +23,7 @@ import kafka.utils.TestUtils._ import kafka.api.FetchRequestBuilder import kafka.message.ByteBufferMessageSet import java.io.File +import java.net.UnknownHostException import kafka.log.LogManager import org.apache.kafka.clients.producer.{KafkaProducer, ProducerRecord} @@ -130,7 +131,7 @@ class ServerShutdownTest extends ZooKeeperTestHarness { val newProps = TestUtils.createBrokerConfig(0, zkConnect) newProps.setProperty("zookeeper.connect", "some.invalid.hostname.foo.bar.local:65535") val newConfig = KafkaConfig.fromProps(newProps) - verifyCleanShutdownAfterFailedStartup[IllegalArgumentException](newConfig) + verifyCleanShutdownAfterFailedStartup[UnknownHostException](newConfig) } @Test diff --git a/core/src/test/scala/unit/kafka/zookeeper/ZooKeeperClientTest.scala b/core/src/test/scala/unit/kafka/zookeeper/ZooKeeperClientTest.scala index 77e11eae716..4eb9e67bf60 100644 --- a/core/src/test/scala/unit/kafka/zookeeper/ZooKeeperClientTest.scala +++ b/core/src/test/scala/unit/kafka/zookeeper/ZooKeeperClientTest.scala @@ -16,6 +16,7 @@ */ package kafka.zookeeper +import java.net.UnknownHostException import java.nio.charset.StandardCharsets import java.util.UUID import java.util.concurrent.atomic.AtomicBoolean @@ -57,7 +58,7 @@ class ZooKeeperClientTest extends ZooKeeperTestHarness { System.clearProperty(JaasUtils.JAVA_LOGIN_CONFIG_PARAM) } - @Test(expected = classOf[IllegalArgumentException]) + @Test(expected = classOf[UnknownHostException]) def testUnresolvableConnectString(): Unit = { new ZooKeeperClient("some.invalid.hostname.foo.bar.local", -1, -1, Int.MaxValue, time, "testMetricGroup", "testMetricType").close() diff --git a/gradle/dependencies.gradle b/gradle/dependencies.gradle index 2ac496b4ca9..b7a03dcd752 100644 --- a/gradle/dependencies.gradle +++ b/gradle/dependencies.gradle @@ -71,7 +71,7 @@ versions += [slf4j: "1.7.25", snappy: "1.1.7.1", zkclient: "0.10", - zookeeper: "3.4.11", + zookeeper: "3.4.10", jfreechart: "1.0.0", mavenArtifact: "3.5.2"] -- This is an automated message from the Apache Git Service. To respond to the message, please log on GitHub and use the URL above to go to the specific comment. For queries about this service, please contact Infrastructure at: users@infra.apache.org