A method apparatus and media for distributing protected content. An exemplary embodiment, including receiving a selection of content, packaging the selected content to generate packaged content, encrypting the packaged content to generate an encrypted content package, transmitting the encrypted content package to a content server, receiving content location information corresponding to the location of the encrypted content package on the content server, receiving a selection of one or more rights, combining the one or more rights with the content location information to generate a rights package including the content location information, transmitting the rights package to a license server, receiving license location information corresponding to the location of the rights package on the license server, and distributing the license location information.
Sending Application

1. Receive a selection of content
2. Package content
3. Transmit content to content server
4. Encrypt content
5. Receive content location info from content server
6. Receive selection of one or more rights
7. Transmit the rights package to a license server
8. Combine one or more rights w/ the content location info to generate a rights package
9. Receive license location info corresponding to the rights package
10. Distribute the license location info

Fig. 2
Player Application

301 Receive license location information corresponding to the rights package and including content location information

302 Transform the license location information to generate transformed license location information

303 Transmit transformed license location information to a license server

304 Receive a usage license from the license server including content location information and usage rights

305 Retrieve content based on the content location information

306 Render the content in accordance with the usage rights

Fig. 3A
Original URL:
www.example.com/licenses/sample_license

Query String Parameters Flag:
"?"

Query String Parameters:
"color=red&size=medium"

Expiration Date:
"&Expires=1357034400"

Signature:
"&Signature=sample~sig"

Key-Pair ID:
"&Key-Pair-Id=SampleID"

Transformed URL:
www.example.com/licenses/sample_license?color=red&size=medium&Expires=1357034400&Signature=sample~sig&Key-Pair-Id=SampleID

Fig. 3B
License Server

1. Receive a rights package including content location information
   - 401

2. Generate license location information corresponding to the rights package
   - 402

3. Receive transformed license location information from a requester
   - 404

4. Transmit the license location information
   - 403

5. Authenticate the requester based on the transformed license location information
   - 405

6. Generate a usage license including the content location information and one or more usage rights
   - 406

7. Transmit the usage license to the requester
   - 407

Fig. 4
Receive a rights package including content location information

Generate license location information corresponding to the right package

Transmit the license location information

Receive the license location information from a requester

Transmit information to requester regarding content and instructions for installing an approved player

Redirect the requester to an approved web client

Receive transformed license location information from the web client

Transmit the usage license to the web client

Generate a usage license including the content location information and one or more usage rights

Authenticate the web client based on the transformed license location information

Fig. 5
Content Server

Receive an encrypted content package
601

Generate content location information corresponding to the encrypted content package
602

Transmit the content location information
603

Receive the content location information from a requester
604

Transmit the encrypted content package to the requester
605

Fig. 6
DISTRIBUTING PROTECTED CONTENT

RELATED APPLICATION DATA

FIELD OF THE INVENTION
[0002] The disclosure relates a method, apparatus and media for federated control of content.

BACKGROUND
[0003] Digital rights management (DRM) enables the delivery of content from a source to a recipient, subject to restrictions defined by the source concerning use of the content. Exemplary DRM systems and control techniques are described in U.S. Pat. No. 7,073,199, issued Jul. 4, 2006, to Raley, and U.S. Pat. No. 6,233,684, issued May 15, 2001, to Stelik et al., which are both hereby incorporated by reference in their entireties.

[0004] One of the biggest challenges with controlling access and use of content is to prevent users from accessing the content in a manner other than those permitted by usage rights. As known in the art, usage rights indicate how content may be accessed or otherwise used. Usage rights may be embodied in any data file and/or defined using program code, and may further be associated with conditions that must be satisfied before access or other use of the content is permitted. Usage rights, which are also referred to as “usage rules” herein, may be supported by cohesive enforcement units, which are trusted devices that maintain one or more of physical, communications and behavioral integrity within a computing system.

[0005] For example, if the recipient is allowed to create an unauthorized (not DRM-protected) copy of the content, then the recipient’s use of the copy would not be subject to any usage restrictions that had been placed on the original content. When DRM systems impose restrictions on the use of a rendering device, for example, by preventing or impeding the use of screen capture features, a conflict arises between the rendering device owner’s (user, receiver, or recipient) interest in being able to operate their device with all of its features without restriction (including screen capture capability), and the content provider’s (sender, or source) interest in regulating and preventing copying of the content rendered on the recipient’s devices. This conflict has historically been overcome by establishing trust between the content supplier and the rendering device. By establishing trust in this manner, the content supplier can ensure that the rendering device will not bypass DRM restrictions on rendered content.

[0006] In contrast, users want flexibility to share content over various proprietary and standard systems. For example, users share content over various social networks and other systems. Some of that content might be protected with usage rights or require proprietary rendering engines or other components. Improvements in content rights management are needed which allow users to create and distribute rights for a variety of content formats and ecosystems for ephemeral (e.g., limited duration) content.

[0007] As a specific example, a user on Facebook might want to post content that is ephemeral and protected. Facebook might then allow other ecosystems like Twitter to render the content in their environment while the content is still under the control of Facebook. Another example of this problem is embedding of YouTube content. YouTube encourages users to share content through Facebook, but it requires an embedded YouTube player to be used inside Facebook.

BRIEF DESCRIPTION OF THE DRAWINGS
[0008] FIG. 1 is a system diagram of a rights generation and distribution system according to an exemplary embodiment.
[0009] FIG. 2 is a flowchart of a method for distributing protected content according to an exemplary embodiment.
[0010] FIG. 3A is a flowchart of a method for accessing protected content according to an exemplary embodiment.
[0011] FIG. 3B is a flowchart of a method for transforming license location information according to an exemplary embodiment.
[0012] FIG. 4 is a flowchart of a method for distributing rights associated with content according to an exemplary embodiment.
[0013] FIG. 5 is another flowchart of a method for distributing rights associated with content according to an exemplary embodiment.
[0014] FIG. 6 is a flowchart of a method for delivering protected content according to an exemplary embodiment.
[0015] FIG. 7 is a schematic block diagram of an exemplary computing environment that may be used to carry out the disclosed methods according to an exemplary embodiment.
[0016] FIG. 8 is a schematic block diagram of an exemplary network environment that may be used to carry out the disclosed methods according to an exemplary embodiment.

DETAILED DESCRIPTION
[0017] This disclosure describes aspects of embodiments for carrying out the inventions described herein. Of course, many modifications and adaptations will be apparent to those skilled in the relevant arts in view of the following description in view of the accompanying drawings and the appended claims. While the aspects of the disclosed embodiments described herein are provided with a certain degree of specificity, the present technique may be implemented with either greater or lesser specificity, depending on the needs of the user. Further, some of the features of the disclosed embodiments may be used to obtain an advantage without the corresponding use of other features described in the following paragraphs. As such, the present description should be considered as merely illustrative of the principles of the present technique and not in limitation thereof.

[0018] Certain implementations of the invention allow a user to associate rights with content, independent of the content platform, and allow other users to access the content in accordance with the associated rights.

[0019] Aspects of the disclosed embodiments address preventing circumvention (e.g., via screen capture) of content subject to digital rights management ("DRM") running on computing platforms. The exemplary embodiments significantly improve the content sender's ability to regulate use of content after the content is distributed.
For the sake of convenience, this application refers to unmodified (e.g., not obscured or censored) content sent by the sender’s device as “source content.” Source content may be encrypted, compressed, and like, and multiple copies of the source content (each copy also referred to as source content) may exist. In addition, content, as disclosed herein, refers to any type of digital content including, for example, image data, video data, audio data, textual data, documents, and like. Digital content may be transferred, transmitted, or rendered through any suitable means, for example, as content files, streaming data, compressed files, etc., and may be persistent content, ephemeral content, or any other suitable type of content.

Ephemeral content, as used herein, refers to content that is used in an ephemeral manner, e.g., content that is available for use for a limited period of time. Use restrictions that are characteristic of ephemeral content may include, for example, limitations on the number of times the content can be used, limitations on the amount of time that the content is usable, specifications that a server can only send copies or licenses associated with the content during a time window, specifications that a server can only store the content during a time window, and the like.

Screen capture is a disruptive technology to ephemeral content systems. It allows the content to persist beyond the ephemeral period (e.g., it allows ephemeral content to become non-ephemeral content). SnapChat, for example, is a popular photo messaging app that uses content in an ephemeral manner. Specifically, using the SnapChat application, users can take photos, record videos, and add to them text and drawings, and send them to a controlled list of recipients. Users can set a time limit for how long recipients can view the received content (e.g., 1 to 10 seconds), after which the content will be hidden and deleted from the recipient’s device. Additionally, the SnapChat servers follow distribution rules that control which users are allowed to receive or view the content, how many seconds the recipient is allowed to view the content, and what time period (days) the SnapChat servers are allowed to store and distribute the content, after which time SnapChat servers delete the content stored on the servers.

Aspects of the disclosed embodiments enable the use (including rendering) of DRM-protected content while frustrating unauthorized capture of the content (e.g., via screen capture), and while still allowing the user (recipient) to visually perceive or otherwise use the content in a satisfactory manner. This is particularly useful when the content is rendered by a DRM agent on a recipient’s non-trusted computing platform. This may be achieved through the application of an obscuration technique (OT) that obscures part or all of the content when the content is rendered. With respect to ephemeral content, obscuration is an enabling technology for ephemeral content systems in that it thwarts a set of technologies that would circumvent the enforcement of ephemeral content systems. The techniques described herein have been proven through experimentation and testing, and test results have confirmed the advantages of the results.

An obscuration technique may be applied during creation of the content or at any phase of distribution, rendering or other use of the content. For example, the obscuration technique may be applied by the sender’s device, by the recipient’s device, by a third-party device (such as a third-party server or client device), or the like. When an obscuration technique (OT) is applied to content during its creation or distribution (e.g., by an intermediate server between the content provider and the end user), the resulting content may be referred to as “obscured content.” When an obscuration technique is applied during the rendering of content the resulting content may be referred to as “obscured rendering” or the resulting rendered content as “obscuredly rendered content.” In addition, the application of an obscuration technique may include the application of more than one obscuration technique. For example, multiple obstructions may be applied during an obscured rendering, either simultaneously or using multi-pass techniques. Thus, the exemplary obscuration techniques described herein may be applied in combination, with the resulting aggregate also being referred to as an obscured rendering.

While aspects of the disclosed embodiments relate to the obscuration technique applied to source content, the obscuration techniques may instead be applied to content in general. For example, the obscuration may be applied to censored content or applied to the rendering of censored content. “Censored content,” as used herein, refers to content that has been edited for distribution. Censored content may be created by intentionally distorting source content (or other content) such that, when the censored content is displayed, users would see a distorted version of the content regardless of whether a user is viewing an obscured rendering or an unobscured rendering of the censored content. Censored content may include, for example, blurred areas. The content may be censored using any suitable means, and censored content may be displayed using a trusted or non-trusted player.

Regarding obscured rendering, aspects of the disclosed embodiments take advantage of the differences between how computers render content, how the brain performs visual recognition, and how devices like cameras capture content rendered on a display. Embeddings of the invention apply obscuration techniques to a rendering of content in a manner that enables the content to be viewed by the user with fidelity and identifiability, but that degrades images created by unwanted attempts to capture the rendered content, e.g., via screen capture using a camera integrated into a device containing the display or using an external camera. As an example, identifiability may be quantified using the average probability of identifying an object in a rendering of content. The content may be degraded content, obscuredly rendered content or source content. At one end of the identifiability score range would be the identifiability score of a rendering of the source content, whereas the other end of the range would be the identifiability score of a rendering of a uniform image, e.g., an image with all pixels having the same color. The uniform image would provide no ability to identify an object.

The identifiability score of the obscuredly rendered content would fall between the scores of the degraded content and the source content, whereas the identifiability score of the degraded content would fall between the scores of the uniform image and the score of the obscuredly rendered content. The average probability of identifying the object in content may be determined as an average over a sample of human users or over a sample of computer-scanned images using facial or other image recognition processes and the like. As an example for fidelity, fidelity may be quantified by comparing the perceived color of one or more regions in rendered degraded content with the perceived color of the one or more regions in the rendered original content, where deviations of the color may be measured using a distance metric in color space, e.g., CIE XYZ, Lab color space, etc. As another example regarding a fidelity metric see (http://live.ece.utexas.
The degraded images captured in this manner will have a significantly reduced degree of fidelity and identifiability relative to the human user's view of content as displayed in an unobscured rendering or a non-obscured rendering. Embedments of the invention also enable a scanning device, such as a bar code or QR code reader, to use the content in an acceptable manner, e.g., to identify the content being obscuredly rendered, while degrading images created by unwanted attempts to capture the obscurely rendered content.

Computers often render content in frames. When an image is captured via a screen shot or with a camera operating at a typical exposure speed (e.g., approximating the frame rate for the display device, e.g., 20-120 Hz), a single frame of the obscurely rendered content may be captured, which will include whatever obscuration is displayed in that frame of the obscurely rendered content. Alternatively, a screen capture or the like may capture multiple frames depending on exposure speed, but embodiments of the invention nevertheless may apply obscuration techniques that cause images captured in this manner to be degraded such that the resulting images have a significantly reduced degree of fidelity and identifiability relative to a human user's perception (or scanning device's scanning and processing) of the obscurely rendered content. In contrast, for a human user, due to persistence of vision and the way the brain processes images, the user will be able to view or otherwise use the obscurely rendered content perceived over multiple frames with fidelity and identifiability.

Ideally, the user will perceive the obscurely rendered content as identical to an unobscured rendering of the content (whether source content, censored content, etc.). The human user may not always perceive the obscurely rendered content as a perfect replication of the unobscured rendering of content because application of the obscuration technique may create visual artifacts. Such artifacts may reduce the quality of the rendering of the content perceived in the obscured rendering, although not so much as to create an unacceptable user experience of the content. An unacceptable user experience may result if objects in the obscurely rendered content are unrecognizable or if the perceived color of a region in the obscurely rendered content deviates from the perceived color of the region in the rendered source content by a measure greater than what is typically accepted for color matching in various fields, e.g., photography, etc.

When considering which obscuration technique should be used, a content provider or sender may consider how the obscuration technique will affect the user's perception of the obscurely rendered content, and also the effect the obscuration technique will have on how degraded the content will appear in response to an attempt to copy of the content via, e.g., a screenshot. For example, a content provider may want to select an obscuration technique that minimizes the effect the obscuration technique will have on the user's perception of an obscured rendering of content, while also maximizing the negative effects the obscuration technique will have on the degraded content.

To determine how the obscuration technique will affect the display of the content, previews of the obscurely rendered content and the degraded content may be displayed to the user. For non-human scanning devices, the content provider or sender may conduct testing of the ability of the scanning device to use obscurely rendered content (e.g., to identify desired information from the obscurely rendered content) subject to varying parameters, e.g., spatial extent and rate of change of the obscuration.

Thus, in summary, when a content supplier wants to distribute source content, the content may be distributed in any form (source content, censored content, etc.). Embodiments of the invention may apply obscuration techniques that enable authorized/intended users or scanning devices to use the obscurely rendered content or the obscured content in a satisfactory manner, while causing unauthorized accesses of obscured renderings to result in degraded content.

To prevent unauthorized access to content, techniques may be used to improve the content sender's ability to regulate content that is to be rendered on a receiver's device. In some cases, these techniques can prevent the content from being perceivable to an authorized user of the content. For example, as described above, the content may be obscured, blurred, distorted, damaged, or otherwise made less useful to a user.

When a user attempts to render content in a trusted computing environment, for example, using a trusted rendering application, the user may be allowed to access the content, which may be obscured content via an obscured rendering. In these cases, the trusted computing environment can be trusted with access to the content because it is part an ecosystem. Members of the ecosystem have been developed to enforce restrictions on the content (e.g., a DRM client can be part of a trusted computing ecosystem). The ecosystem can be bounded by those clients that can be trusted to follow usage rules associated with content and access to content encryption keys for that content). Content ecosystems today have a rich ability to move content freely across ecosystems. As an example, Facebook can post YouTube videos, email attachments (e.g., images) can be posted on websites, etc. One obstacle to widespread deployment of protected content across ecosystems (e.g., with unprotected content in the examples above) is that the protected content needs to be displayed and transported by a variety of systems, including systems that are not part of the trusted ecosystem for a given piece of protected content (e.g., web browsers). One approach is to have the rendering agent for the website become part of the trusted ecosystem (e.g., have the Facebook page that is sent to a client browser include a jpg rendering routine that is trusted to enforce usage rules associated with the content). An alternative is to use a rendering engine of the browser platform to render the protected content by wrapping it with a DRM agent. The use of exemplary DRM agents are described in U.S. Pat. No. 7,743,259, issued Jun. 22, 2010, and various DRM system and controls techniques (such as those described herein) may be used with the techniques described herein.

Aspects of the disclosed embodiments relate to distributing protected content, including receiving a selection of content, packaging the selected content to generate packaged content, encrypting the packaged content to generate an encrypted content package, transmitting the encrypted content package to a content server, receiving content location information corresponding to the location of the encrypted content package on the content server, receiving a selection of one or more rights, combining the one or more rights with the content location information to generate a rights package, including the content location information, transmitting the rights package to a license server, receiving license location
information corresponding to the location of the rights package on the license server, and distributing the license location information.

[0036] Aspects of the disclosed embodiments relate to accessing protected content according to an exemplary embodiment, including receiving license location information corresponding to a rights package on a license server, the rights package including content location information and one or more rights, wherein the content location information corresponds to a location of an encrypted content package, transforming the license location information to generate a transformed license location information, transmitting the transformed license location information to the license server, receiving a usage license from the license server including the content location information and one or more usage rights, retrieving the encrypted content package based on the content location information, and rendering content in the encrypted content package in accordance with the one or more usage rights.

[0037] Aspects of the disclosed embodiments relate to distributing rights associated with content according to an exemplary embodiment, including receiving a rights package including content location information corresponding to content and one or more rights associated with the content, generating license location information corresponding to the location of the rights package, transforming the license location information, receiving a transformed license location information from a requestor, wherein the transformed license location information is generated from the license location information, authenticating the requestor based at least in part on the transformed license location information, generating a usage license including the content location information and one or more usage rights based at least in part on the one or more rights associated with the content, and transmitting the usage license to the requestor.

[0038] Aspects of the disclosed embodiments relate to distributing rights associated with content according to another exemplary embodiment, including receiving a rights package including content location information corresponding to content and one or more rights associated with the content, generating license location information corresponding to the location of the rights package, transmitting the license location information, receiving the license location information from a requestor, wherein the transformed license location information is generated from the license location information, authenticating the web player client configured to transform the license location information into a transformed license location information, receiving a transformed license location information from the web player client, wherein the transformed license location information is generated from the license location information, authenticating the web player client based at least in part on the transformed license location information, generating a usage license including the content location information and one or more usage rights based at least in part on the one or more rights associated with the content, and transmitting the usage license to the web player client.

[0039] Aspects of the disclosed embodiments relate to delivering content according to an exemplary embodiment, including receiving an encrypted content package, generating content location information corresponding to the location of the encrypted content package, receiving the content location information from a requestor, and transmitting the encrypted content package to the requestor.

[0040] Location information can include any information that identifies a location or which resolves to identify a network location of content or network location of a license. For example, location information can include an identifier, such as a tag or a value, which can be used to look up or create a Uniform Resource Locator (URL) for content or for a license. Additionally, location information can also be a URL. For example, the content location information can be a content URL and the license location information can be a license URL. Location information can also include location information on a specific device, such as a folder or sub-folder.

[0041] Additionally, license location information can identify a parameter that is used to generate a license. For example, license location information can be information that enables a player application to identify a server to transmit a request to, and parameter information that allows the identified server to build or retrieve the information (e.g., license) that is specific to the content that the player application is interested in rendering. An example of the parameter information would be a 128 bit globally unique identifier (GUID) that is used by a server to look up what license should be generated in response to the license request.

[0042] FIG. 1 is a system diagram of a rights generation and distribution system according to some embodiments. The specific steps carried out by each of these components will be described in greater detail with respect to FIGS. 2-6. As illustrated in FIG. 1, the architecture of the system includes a content server 110 and a license server 120. Sending application 130 and player application 140 can reside on respective devices, and are described in detail below. The various devices can communicate over any type of known network, such as the internet.

[0043] As shown in FIG. 1, sending application 130 can transmit selected content to the content server 110. The content server 110 can store the selected content and generate unique content location information, which can be sent back to the sending application 130. The sending application 130 then associates selected rights with the content location information in the form of a rights package and can transmit the rights package, including the selected rights and the content location information, to the license server 120. The content server 110 and the license server 120 can each include multiple devices of course. Furthermore, there can be more than one content server 110 and more than one license server 120 in any embodiment.

[0044] The license server 120 can store the rights package and generate unique location information, which can be sent back to the sending application 130. The sending application can then make the license location information available to other users. For example, the license location information can be sent to a player application 140 running on a user client device, as shown in FIG. 1.

[0045] Alternatively, the sending application 130 can upload content to the content server 110 and send usage rights and the content location information to a distribution server (not illustrated in FIG. 1). Note that the distribution server can be integrated and co-located with content server 110 or can be at a separate location. Usage rights, which are also referred to as usage rules herein, can be supported by cohesive enforcement units, which are trusted devices that can maintain one or more of physical, communications and behavioral integrity within a computing system. The distribution server can then use the rights and content location information from the sending application to generate a rights package as outlined above.

[0046] The player application 140 can transform the license location information and transmit the transformed license location information.
location information to the license server 120. This transformation involves the addition of information to the license location information which can be used to authenticate the player application 140 and is described in greater detail further below with reference to FIGS. 3A-3B. The license server 120 can authenticate the player application based on the transformed license location information and transmit a usage license including the content location information back to the player application 140. The player application 140 can then retrieve the content using the content location information and render the content in accordance with the usage license.

Fig. 2 illustrates an example method of operation of the sending application 130 and is a flowchart associated with distributing protected content according to some embodiments. At step 201 a selection of content is received, e.g., from a user client device running the sending application 130. This can include selection of that is stored on the user device that is running the sending application 130 or external content that is stored remotely. For example, a user can select a video file that is being hosted by a remote file hosting service and the video file can then be downloaded to the device or a link to the video file can be used to reference the file. Content can include any type of digital content, for example, multimedia, social networking, or textual information, such as videos, pictures, Twitter tweets, documents, spreadsheets, PowerPoint presentations, portable document files (PDFs), text, web pages, etc. The content can also be content created by the user, such as a video captured on the same user mobile device that is running the sending application. Furthermore, the content can be source content, censored content, degraded content, obscured content, and the like.

At step 202 the content can be packaged in an archive format to create packaged content, such as by using a UNIX archive format, a gzip format, a tape archive format, or the like. The compression format can also be a royalty-free format with a broad array of compress/decompress platform support such as iOS/Android/PC/Mac/Linux. The content package can include a payload file, such as a single self-contained archive file that can be unpacked by the player application to recover multiple files. The content package can be agnostic of file type for the payload files. The payload can include a varying number of files, depending on the content. For example, payload can include 100 files or 20 files. The file size of each of the files and the total payload size can also vary. The payload can also include a globally unique identifier (GUID), which can be generated at the time of packaging.

At step 203 the packaged content can be encrypted to generate an encrypted content package. The packaged content can be encrypted using a local, randomly-generated symmetric key. For example, a segment of the packaged content can be encrypted based on an ecosystem-wide (including the sending application and player application) single secret or other code or the like. The single secret can be unique to each version of the ecosystem (e.g., V1 (version 1) of the package uses Secret 1 etc.). The secret can be stored in encrypted format as part of the payload.

A portion of the package can be left unencrypted so that non-trusted components can help manage the content package lifecycle. For example, the unencrypted portion of the package can have clear (unencrypted) metadata such as the GUID (same as the GUID stored in the payload), a workflow or application identifier, a package version, a copyright, or a uniform resource locator (URL) for a compatible player to access the content. Alternatively, the entire content package can be left unencrypted and then encrypted later by a content server that receives the package. In this scenario, the encryption keys can be made accessible to the content server.

At step 204 the content package (including the encrypted and unencrypted portions) can be sent to a content server. This can be accomplished by sending the content package via a transport mechanism, such as email, uploading to the cloud, or removable media. In response to transmitting the content package to the server, the content server stores the content package and returns unique content location information corresponding to the transmitted content package.

At step 205 the sending application can receive and record the content location information and any other information provided by the content server regarding how a recipient can gain access to the content package. At step 206 a selection of one or more rights is received. Of course, this step can be performed prior to any of the earlier steps. For example, a selection of one or more rights can be received at the same time that a selection of content is received, such as through a user interface which allows a user to select content and specify one or more rights.

When receiving a selection of rights, the sending application can display options corresponding to usage rights which determine how the protected content can be used and can also display options corresponding to distribution rights which determine how the protected content can be distributed. The rights can specify who may exercise the rights and the content package to which the rights are assigned. The rights can include usage and distribution rights indicating what can be done with the packaged content and conditions specifying under what limitations the rights can be exercised. For example, the rights can specify a principal user and password, a GUID of a payload, display rights, and extraction rights. Conditions can include application conditions such as limiting rights to the first player application to open the content or time window conditions which indicate a duration in which the rights must be exercised. Additionally, the conditions can include geographic perimeters or conditions, such as a condition that the rights must be exercised within 50 miles of an identified city.

Distribution rules regulate which entities are entitled to usage rights and when and under what conditions a distributor application, such as a distributor application running on the license server, is authorized to distributed usage rights. Conversely, usage rights regulate how and when user of content is allowed to use content that has been distributed by the distribution application. An example of a distribution right would be that server ABC is allowed to distribute rights to content XYZ during time window D, to entities E, F, and G. An example of an usage right would be that entities E, F, and G are allowed to view the content one time during time window H.

As described above, according to aspects of the embodiments, the content may be ephemeral content. Some examples of conditions that can apply to rendering or storage of content that is used in an ephemeral manner include: content only playable a limited number of times; content only playable for a limited number of seconds; server can only
send copies or licenses during a limited time window; server can only store content for a limited number of days; server may be required to eliminate keys to content immediately after encrypting content (the keys may be encrypted using another key and then deleted, which allows the server to encrypt content but not have access to the keys for future use); server can only distribute content and/or keys to identified recipients; content can only be displayed with an obscuration technique applied; content and/or keys can only be distributed to trusted applications, wherein trusted applications are authenticated to enforce the rules associated with the content; content and/or keys can only be distributed to other trusted distributors; content and/or keys can only be distributed to identified distributors; only identified entities can modify the rights to the content (conditions/rules); only authenticated trusted entities can modify the rights to the content (conditions/rules); and/or trusted applications must check and/or set the state of conditions, before and/or during and/or after, rendering (this enables accounting of ephemeral content, as well as real-time control like a “kill” switch that allows the sender to disable use of content after distribution, resulting in revocation).

[0056] Conditions that relate to the storage of the content on the content server 110 can be enforced by the license server 120. For example, if a condition specifies that the content can only be stored for a predetermined period of time on the content server 110, then after expiration of the predetermined period of time, the license server 120 can send an instruction to the content server 110 to delete the content. Any conditions relating to the content server can be sent to the license server 120 by the sending application 130 as part of a rights package so that it can enforce the conditions.

[0057] Alternatively, such distribution conditions can be enforced through the use of buckets, or content storage categories, on the content server. When the content is transmitted to the content server 110, a bucket can be specified which is associated with a predetermined set of actions and the predetermined set of actions can be performed on all content stored in that bucket. For example, a first bucket can specify that all content added to that bucket is deleted after five days and a second bucket can specify that all content added to that bucket is deleted after a month but that the content added can only be transmitted fifty times. Many variations of conditions are possible for each bucket and these examples are not intended to be limiting.

[0058] In this context the term “bucket” is an identifier that can be used to determine how to treat the content being sent to the content server 110. The server can store parameters associated with the identifier and can use the identifier to associate information with the content as it is received. For example, an identifier (bucket) can be an “ephemeral message” identifier. The content server 110 can then treat all content received that is associated with the identifier “ephemeral message” as having an associated rule: “delete 20 minutes after receipt.” The content server 110 can also modify any rules associated with a given identifier. Additionally the identifier (bucket) can be used to assign distribution rules, usage rules or both. The use of exemplary distribution rules and usage rules are described in U.S. Pat. No. 7,743,259, issued Jun. 22, 2010, and various DRM system and controls techniques (such as those described therein) can be used with the techniques described herein.

[0059] At step 207 the sending application 130 combines the selected rights information and the content location information into a rights package to transmit to a license server 140. The rights package can be a data structure, as set forth below, stored on non-transient media. The encoding of this information can be versioned so that the license server 140 (or the distribution server) is capable of interacting with multiple sending applications that have varying functionality. For example, the encoding of the rights package can be customized for an email application, for Twitter, for a messaging application, or for a broadcast application. In addition to the rights information and the content location information, the rights package can include the ecosystem type (e.g., content player for Twitter), the symmetric content key, and/or the sender ID (such as an install ID or user ID). For example, a rights package could look like:

```xml
<Sample Rights Package>
<Content Location Information = "www.server.com/contentid=212412121">
<Distribution Rights>
 <Distribution Duration = 10 days>
 <Distribution Authentication Type = password protection>
 <Distribution encrypted password = dsadasaw>
 <Distribution Encryption Key ID = 23>
 <Distribution Rights>
 <Usage Rights>
 <Usage Duration = 1 day>
 <Reproduction Permitted = false>
 <Usage Rights>
</Sample Rights Package>
```

[0060] As discussed earlier, the sending application may also transmit the selected rights information and the content location information to a distribution server (or a distribution application) instead of a license server. The distribution server may combine the selected rights information and the content location information into the rights package. Additionally, instead of receiving content location information from a content server, the content may be sent to a content server which utilizes the buckets or identifiers described above to associate distribution rules and/or usage rules with content based on the associated identifier and which would then function as both a license and a content server.

[0061] At step 208, the sending application 130 transmits the rights package to the license server. The license server then stores the rights package and generates license location information corresponding to the rights package. At step 209, license location information corresponding to the location of the rights package is received at the sending application 130 from the license server. The license location information may be a URL that allows the license server 120 to identify which content license is being requested. As a convenience, if the
license location information is a long URL, then the license location information may be shortened using a mapping algorithm like Goo.gl, bit.ly, or a custom mapping algorithm. For example, the license location information: http://Yovolicensever.yovo.com:packageld:12341234:userid:bobo@bobo.com:expiration:1/1/2015:authtoken:SI/#&$(*&c&$&@((full internal License URL)) may be shortened to http://yovo.co/JDF12. In this case, the sending application may receive only the shortened version.

[0062] At step 210 the sending application 130 distributes the license location information via any appropriate means. The license location information may, for example, be transmitted via push notification in a messaging application or posted to Twitter in a Twitter application. The distributed license location information may either be handled directly by a user in some cases or completely hidden from the user. The license location information may also be transmitted with a censored or partial version of the content.

[0063] FIG. 3A illustrates and example of the operation of the player application 140 and is a flowchart associated with accessing protected content according to some embodiments. At step 301 the player application 140 may receive license location information corresponding to a rights package that includes content location information. The mechanism by which the license location information is received may depend on the particular ecosystem. For example, in a messaging ecosystem, the license location information may be received as a message from the sending application. If the ecosystem is a Twitter ecosystem, then the license location information may be obtained by browsing another user’s feed. A user of the player application may also be notified that protected content has been received.

[0064] Additionally, when the user receives the license location information, they may also receive the partial or censored version of the content, which may be rendered on the player application. Additionally, if the user attempts to access the license location information without the appropriate application or an unauthorized application or web client, then the application may present the censored or partial version of the content, instead of an uncensored or complete version of the content. As an example, the license location information may be a fully formatted URL that is processable by a standard web browser. If a recipient of the license location information were to use a standard web browser to open the URL, the license server would receive the request. Because the license location information has not been transformed as outlined below, the server may use techniques (e.g. HTTP-USER-AGENT as provided by the browser) to determine that a browser, which may be an untrusted application, has made the request. In response to the request the license server may redirect the client to a web server that acts as a trusted player (web player), in which case the web player may transform the request and resubmit it to the license server and then send a browser view of the content (with usage rules enforcement) to the requesting browser. Alternatively, the license server may generate a page informing the requesting browser user that they should open the URL using a trusted client. This page may optionally include a censored view of the content to encourage the requester to obtain the trusted client.

[0065] At step 302, the player application 140 may transform the license location information in such a way that the license server may identify information necessary to authenticate the player (digital signature) as well as provide versions and ecosystem information to allow the license server to provide appropriately formatted usage licenses. For example, the transformation performed on license location information for a twitter ecosystem may be different than the transformation performed on license location information for multimedia video sharing ecosystem or an office document ecosystem.

[0066] FIG. 3B illustrates a flowchart for a method of transforming the license location information according to some embodiments. As shown in the figure, the license location information may be in the form of an original (base) URL. At step 311 a query string parameters flag may be added to the URL by the player application to indicate that query string parameters will follow the base URL. The query string parameters flag delineates the original URL from what is going to be added during the transformation.

[0067] At step 312 query string parameters may be added to the URL after the delineating flag 311. As shown in FIG. 3B, two query string parameters have been added, indicating that the color=red and that the size=medium. The query string parameters may be separated by an ampersand. Query string parameters may be used to add information relating to the player application that is used to play the content, such as the ecosystem, the version, a user ID, an installation ID, the type of player application, and/or a content file type. This information may be utilized by the license server 120 when formatting the usage rights for delivery to the player application.

[0068] At step 313 an expiration date may be added to the URL by the player application. The expiration date may include the date and time of expiration in any time format. FIG. 3B illustrates an expiration date of Jan. 1, 2013 at 10 AM in Unix time format and Coordinated Universal Time (UTC). The expiration date may function as a timer to prevent the transformed URL from being used or reused at a later date.

[0069] At step 314 a signature may be added to the URL. The signature may be a hashed and signed version of a policy statement for a particular resource, such as the license. The signature may be computed from the transformed URL as well. For example, the signature may be generated by computing a hash value from the entire transformed URL up to portion 313. This hash value may then be encrypted using a symmetric key that is available to the license server 120 or alternatively by using a public (asymmetric) key corresponding to a private key held by the license server 120.

[0070] At step 315 a key pair ID for the key pair used in the encryption process for the signature may be added to the URL. This key pair ID may be used by the license server when evaluating the transformed license location information (the transformed license URL in this example). In other words, the key pair ID may notify the license server 120 of which key to utilize when decrypting the signature. The result of these steps is a transformed license URL, shown in FIG. 3B, which includes the portions added during the transformation process and which can be stored as a data structure on a non-transient media.

[0071] Returning to FIG. 3A, at step 303 the player application 140 may determine the license server 120, from among plural license servers, identified by the license location information and transmit the transformed license location information to the license server 120. For example, in the scenario where the license location information is a URL, the player application 140 may identify the appropriate license server based on the URL and send a transformed URL to the identified license server. After the license server 120 uses the
transformed license location information to authenticate the player application 140 (this authentication is explained further with reference to FIG. 4), a usage license including the content location information and usage rights is received by the player application 140 at step 304. As discussed earlier, the usage license will be appropriately formatted to the player application 140 and ecosystem since the transformation performed by the player application 140 on the license location information alerts the license server 120 to the appropriate format to use.

[0072] The usage license contains information unique to the version and ecosystem of the player application that allows that player application to retrieve the content package identified by the content location information. The player application 140 retrieves the content package at step 305 and may cache it locally. The player application 140 is responsible for managing the cache based on platform and requirements of the ecosystem and version of the player. For example, when the player application 140 receives license location information, it may initiate usage license and content package requests in the background without notifying the user.

[0073] At step 306 the player application 140 renders the content in the content package in accordance with the usage license. The content package may be decrypted using the shared secret for the particular version of the ecosystem and player application. The player application 140 may enforce any usage license restrictions prior to, during, and after rendering of the content. In addition, as required by the usage license or ecosystem, the player application 140 may report status and accounting information to the license server 120 as identified by the license location information (e.g. screenshot taken, played, deleted, location of user or player application, etc.).

[0074] FIG. 4 illustrates and example of the operation of the license server 120 and is a flowchart associated with distributing rights associated with content according to some embodiments. At step 401 a rights package including content location information corresponding to content and one or more rights associated with the content is received. The rights package may be received from the sending application 130 as described earlier.

[0075] Additionally, a bidirectional trusted session may be established between a sending application 130 and the license server 120 (the license server 120 authenticates the sending application, and vice versa). For example, a trusted session may be established using Secure Sockets Layer (SSL) and/or Transport Layer Security (TLS). These technologies allow a server and client to securely exchange a “session key.” The exchange of messages sent between the client and the server using this session key would be referred to as a “trusted session”. A trusted session may also include technologies such as quantum entanglement and quantum cryptography, which allow for a communication channel between a client and a server that allows each to receive messages that only intended recipients are allowed to access.

[0076] As discussed earlier, the rights package is a data structure and may include an ecosystem type, a version, content location information, a symmetric content key used to protect a content package, distribution rights and usage rules, and a sender ID such as an install ID or user ID.

[0077] The rights package may be stored at the license server 120 or an attached storage and at step 402 license location information corresponding to the location of the rights package is generated by the license server 120. The license location information may be a unique URL corresponding to the received rights package that allows a requester to request a license to the content associated with content location information. For example, the unique license location information may be: http://yoyo.co/JDFJ12. Additionally, the rights package may be hosted by multiple different license servers so that a subsequent user or application is not constrained to accessing the rights package through a single server. Multiple license servers may also be used for different types of rights packages. For example, a first license server may be used to host rights packages relating to a first ecosystem or application type, and a second license server may be used to host rights packages relating to a second ecosystem or application type.

[0078] At step 403 the license location information is transmitted to the sending application 130. At step 404 a transformed license location information is received from a player application 140, the transformed license location information being generated from the license location information. Optionally, non-transformed license location information may be received at the license server 120 as well. This scenario is explained in greater detail with reference to FIG. 5 below. To deal with both contingencies, a determination may be made when a transmission is received regarding whether the received transmission includes transformed license location information or non-transformed license location information.

[0079] When a transformed license location information is received, the player application 140 is authenticated based at least in part on the transformed license location information at step 405. The license server 120 uses the transformed license to authenticate the player application 140, including, for example, the ecosystem of the player application 140, the version of the player application 140, a user ID of a user using the player application 140, an installation ID of the player application 140, etc.

[0080] For example, if the transformed license information is the transformed URL as described in FIG. 3B, then the license server 120 may identify the appropriate key to decrypt the signature based on the key pair ID and may generate the hash value computed by the player application 140 by decrypting the signature. The license server 120 may compute a second hash value from the transformed URL using the appropriate hash function and may compare the decrypted hash value to the generated second hash value to determine if they match.

[0081] If the authentication indicates that the player application 140 is not a valid player, the license server 120 may send a response indicating to the player application 140 what steps are necessary to receive a proper license. The transformed license location information may also be used to indicate to the license server 120 the type of player application 140 and content file type so that the usage license may be formatted appropriately. For example, a certain application type or ecosystem may allow for implementation of certain usage rights or distribution rights which are not possible on a different application or ecosystem.

[0082] If the authentication indicates that the player application 140 is valid, a usage license including the content location information and one or more usage rights is generated at step 406 based at least in part on the one or more rights associated with the content. Since the license server may be used to serve multiple application ecosystems simultaneously
the information recorded in the rights package may be formatted and versioned by the license server to a usage license that is compatible with the player application. In this model the authorization to render the content is carried out by the player application evaluating the usage rules and controlling the content accordingly.

[0083] In order to deliver a well-formatted usage license for multiple different formats and/or application ecosystems, the license server 120 may store the usage rights as a default or intermediate representation and then format the intermediate representation based on the particular application ecosystem. The license server 120 may store format information for a varied selection of different application ecosystems and then use that format information to modify the intermediate representation in order to customize the transmitted usage rights or usage license to the appropriate format for the player application 140. For example, if a usage right relating to permitted video resolutions specifies that only resolutions up to 800x600 are permitted and if a particular player application includes settings only for discrete resolutions 640x400 and 1600x1200, then the intermediate representation of the usage right may be modified to specify a resolution of 640x400 since that is the highest possible resolution that may be displayed by the player application 140 within the bounds of the usage right.

[0084] Additionally, the license server 120 may request information from the player application 140 (location, time, user ID, etc.) and evaluate the usage rules to authorize playback. In this case a much simpler set of rules is included in the usage license. In other words, the license server 120 may be used to enforce certain rights associated with the content as distribution rights, rather than transmitting the rights to the player application 140 for enforcement as usage rights. For example, if one of the distribution rights specifies that only users within 10 miles of a certain location may access the content, the license server 120 may query the player application 140 to determine whether the user is within 10 miles of the location. If the user fulfills this requirement, the subsequent usage license may omit the 10 miles requirement. Conversely, if the user does not fulfill the requirement, then the process may end without generation of a usage license.

[0085] At step 407 the generated usage license is transmitted by the license server 120 to the player application 140. The usage license may include usage rules such as time windows, geo location information, identity information, etc. The usage license may also include content keys, the content location information, the contact information for a sender, a user of the sending application 130, or a creator of the content. Additionally, the license server 120 may encrypt the usage license so that only an authenticated requestor or an authenticated player application may access it. Furthermore, the generated usage license may be based on the type of content which it relates to and the type of file which is being rendered by the player application 140.

[0086] In addition, the license server 120 may maintain accounting information about when licenses are issued and where each usage license is delivered. Furthermore, if the usage rules specify accounting for information to be reported from the player application 140 (such as advertisements viewed, content rendered, total number of content renderings, duration of content rendered, user location, or any other requirements or conditions), the license server 120 has an application programming interface (API) that allows player applications to report such information.

[0087] FIG. 5 illustrates an example of the operation of the license server 120 and is a flowchart associated with distributing rights associated with content when the end user attempts to access content through an unauthorized application, according to some embodiments. At step 501 a rights package including content location information corresponding to content and one or more rights associated with the content is received. The rights package may be received from the sending application 130 as described earlier. Additionally, a bidirectional trusted session may be established between a sending application 130 and the license server 120 (the license server 120 authenticates the sending application, and vice versa). As discussed earlier, the rights package may include an ecosystem type, a version, content location information, a symmetric content key used to protect a content package, distribution rights and usage rules, and a sender ID such as an install ID or user ID.

[0088] The rights package may be stored and at step 502 license location information corresponding to the location of the rights package is generated. The license location information may be a unique URL corresponding to the received rights package that allows a requestor to request a license to the content associated with content location information. For example, the unique license location information may be: http://yoyo.co/JDF/12.

[0089] At step 503 the license location information is transmitted to the sending application. At step 504 the license location information is received from a requestor. Since the license location information is not transformed and is a copy of the license location information transmitted to the sending application, the license server 120 infers that a browser has attempted to load the license location information.

[0090] In this case, many responses are available. Optionally, at step 505, the license server 120 may look up the rights information and metadata associated with the license location information and return information (such as an HTML page) which gives details about the content and how the user may install an application that will allow full access to the content, such as an approved player application 140. Additionally, the web browser may present the censored or partial version of the content.

[0091] Alternatively, at step 506, the license server 120 may redirect the requestor to an approved web player client (which may be an optional feature for particular ecosystems). Processing would then proceed similar to that of FIG. 4, with the web player client taking the place of the player application 140.

[0092] Specifically, at step 507, transformed license location information is received from the web player client, the transformed license location information being generated from the license location information. When transformed license location information is received, the web player client is authenticated based at least in part on the transformed license location information at step 508. If the authentication is valid, a usage license including the content location information and one or more usage rights is generated at step 509 based at least in part on the one or more rights associated with the content. At step 510 the generated usage license is transmitted to the web player client. The subsequent access of the protected content on the web player client may then be rendered in the requestor's browser.

[0093] FIG. 6 illustrates an example of the operation of the content server and is a flowchart associated with delivering content according to some embodiments. The content server
does not have to be a trusted entity of the system, because the packages are encrypted and the content server does not need to see the internals of the packages. However, the content server may authenticate agents placing content in the server in order to protect the server from being abused by non-trusted entities. Additionally, the content server may perform garbage collection to reduce wasted space by packages that are no longer eligible to be accessed. Furthermore, since the content server does not need to see the internals of any of the packages, the content server may be utilized to host multiple different content types on a single server or set of servers.

At step 601 an encrypted content package is received, such as from the sending application. For example, the content server may receive a request to ingest a single content file (such as an archive file containing multiple archived files) that has been encrypted with a key that is unknown to the content server. In addition the content server may receive information that may help manage garbage collection and may authenticate the ingest request as a trusted entity of the system to minimize possible abuse of storage and bandwidth.

Alternatively, an unencrypted content package may be received and then encrypted by the content server using a key that is provided to the content server. For example, the encryption key may be sent in a secured form along with the unencrypted content package, or the key may be stored at the content server rather than at the sending application.

At step 602 the encrypted content package is stored and content location information corresponding to the location of the encrypted content package is generated. If the request to ingest the content is permitted and successful, the return value is content location information, which is transmitted at step 603. The content location information is information that a player may deliver to the content server to recover the encrypted content package.

At step 604 the content location information from a requestor is received. The requestor may be a player application 140 or web client that contacts the content server to request a distribution or download of the encrypted content package. Optionally, the content server may authenticate the request to avoid possible abuse of bandwidth by overloading the content server with bogus download requests. This authentication may be similar to the authentication performed by the license server 120. For example, the content location information may be transformed and then the transformed content location information may be sent to the content server and authenticated by the content server. An alternative to authenticating the download request may be to provide monitoring/detection and response of aberrant download requests.

If authentication is successful, or if no authentication is performed, then at step 605 the encrypted content package is transmitted to the requestor.

The obscuration techniques described herein can be applied to content in a variety of ways. In some embodiments, the following process may be used. First, an image layer can be created for the obscured rendering. This image layer may include the source content (or any other content to be displayed). If a masking obscuration technique is being used, a mask layer can also be created, which may accept user interface elements. This layer can be overlaid over the image layer in the display. The mask layer can be any suitable shape, for example, a circle, a square, a rounded corner square, and the like. During rendering, the mask layer should not prevent the image layer from being viewed unless there are obscuration elements within the mask layer that obscure portions of the image layer. In some embodiments, the mask layer can be configured by a content owner or supplier through any suitable input method, for example, by touching, resizing, reshaping, and the like. Then, one or more sequence of images can be created from the source content, and each image in each sequence can be a transformation of the source content. When the sequences of images are viewed sequentially, for example, at the refresh rate of the display screen or at a rate that is less than the refresh rate of the display screen (e.g., every other refresh of the screen, etc.), the displayed result of the sequences of the images approximates the original source image. In some embodiments, multiple sequences of image frames (e.g., 2-100 or more in a sequence) can be generated, and more than one type of transformation technique may be used. The image frames from one or more of the sequences can then be rendered at a rate that can be approximately the refresh rate of the display screen (e.g., 15-240 Hz). In some embodiments, the user can select which sequence of image frames to display (e.g., sequence 1, sequence 2, etc.).

The mask layer can then be used to overlay the rendered sequence over the image layer, which creates a background of the source image via the image layer with the mask layer selecting where to show the sequence of transformed image frames. In some embodiments, the user can manipulate the mask layer while also previewing different sequences of image frames, and the user can also select a combination of a mask shape and/or form with a selection of a sequence. The resulting selections can be stored, associated with the source content, and distributed with the source content.

The source content and the selected mask and sequence(s) can then be transmitted to a receiving device. When the receiving device renders the source content, the selected mask and the selected sequence of image frames can be used to render the content obscuring.

When obscuration techniques are applied to still images according to some embodiments, the obscuration techniques frames in a frame set may be converted to GIF frames, for example. These GIF frames then can be saved in an animated GIF file format for playback as an n-frame loop.

Another approach takes advantage of computing devices with graphic processors (GPUs) and multiple frame buffers. A frame buffer consists of a large block of RAM or VRAM memory used to store frames for manipulation and rendering by the GPU driving the device’s display. For GPUs supporting double buffering with page flipping, and for still image obscuration techniques with a two-frame cycle, some embodiments may load each obscuration techniques frame into separate VRAM frame buffers. Then each buffer may be rendered in series on the device’s display at a given frame rate for a given duration. For GPUs supporting triple buffering, and for still image obscuration techniques with a two-frame cycle, in some embodiments, each obscuration techniques frame may be loaded into separate RAM back buffers. Then each RAM back buffer may be copied one after the other to the VRAM front buffer and rendered on the device’s display at a given frame rate for a given duration.

One or more of the above-described techniques can be implemented in or involve one or more computer systems. FIG. 7 illustrates a generalized example of a computing environment 700 that may be employed in implementing the embodiments of the invention. The computing environment
700 is not intended to suggest any limitation as to scope of use or functionality of described embodiments.

[0105] With reference to FIG. 7, the computing environment 700 includes at least one processing unit 710 and memory 720. The processing unit 710 executes computer-executable instructions and may be a real or a virtual processor. The processing unit 710 may include one or more of: a single-core CPU (central processing unit), a multi-core CPU, a single-core GPU (graphics processing unit), a multi-core GPU, a single-core APU (accelerated processing unit), combining CPU and GPU features, or a multi-core APU. When implementing embodiments of the invention using a multi-processing system, multiple processing units can execute computer-executable instructions to increase processing power. The memory 720 may be volatile memory (e.g., registers, cache, RAM, VRAM), non-volatile memory (e.g., ROM, EEPROM, flash memory, etc.), or some combination of the two. In some embodiments, the memory 720 stores software instructions implementing the techniques described herein. The memory 720 may also store data operated upon or modified by the techniques described herein.

[0106] A computing environment may have additional features. For example, the computing environment 700 includes storage 740, one or more input devices 750, one or more output devices 760, and one or more communication connections 770. An interconnection mechanism 780, such as a bus, controller, or network interconnects the components of the computing environment 700. Typically, operating system software (not shown) provides an operating environment for software executing in the computing environment 700, and coordinates activities of the components of the computing environment 700.

[0107] The storage 740 may be removable or non-removable, and may include magnetic disks, magnetic tapes or cassettes, CD-ROMs, CD-RWs, DVDs, or any other medium which can be used to store information and which can be accessed within the computing environment 700. In some embodiments, the storage 740 stores instructions for software.

[0108] The input device(s) 750 may be a touch input device such as a keyboard, mouse, pen, trackball, touch screen, or game controller, a voice input device, a scanning device, a digital camera, or another device that provides input to the computing environment 700. The input device(s) 750 may also be incorporated into output device 760, e.g., as a touch screen. The output device(s) 760 may be a display, printer, speaker, or another device that provides output from the computing environment 700.

[0109] The communication connection(s) 770 enable communication with another computing entity. Communication may employ wired or wireless techniques implemented with an electrical, optical, RF, infrared, acoustic, or other carrier.

[0110] Implementations can be described in the general context of computer-readable media. Computer-readable media are any available storage media that can be accessed within a computing environment. By way of example, and not limitation, within the computing environment 700, computer-readable media may include memory 720 or storage 740.

[0111] One or more of the above-described techniques can be implemented in or involve one or more computer networks. FIG. 8 illustrates a generalized example of a network environment 800 with the arrows indicating possible directions of data flow. The network environment 800 is not intended to suggest any limitation as to scope of use or functionality of described embodiments, and any suitable network environment may be utilized during implementation of the described embodiments or their equivalents.

[0112] With reference to FIG. 8, the network environment 800 includes one or more client computing devices, such as laptop 810A, desktop computing device 810B, and mobile device 810C. Each of the client computing devices can be operated by a user, such as users 820A, 820B, and 820C. Any type of client computing device may be included.

[0113] The network environment 800 can include one or more server computing devices, such as 870A, 870B, and 870C. The server computing devices can be traditional servers or may be implemented using any suitable computing device. In some scenarios, one or more client computing devices may function as server computing devices.

[0114] Network 830 can be a wireless network, local area network, or wide area network, such as the internet. The client computing devices and server computing devices can be connected to the network 830 through a physical connection or through a wireless connection, such as via a wireless router 840 or through a cellular or mobile connection 850. Any suitable network connections may be used.

[0115] One or more storage devices can also be connected to the network, such as storage devices 860A and 860B. The storage devices may be server-side or client-side, and may be configured as needed during implementation of the disclosed embodiments. Furthermore, the storage devices may be integral with or otherwise in communication with the one or more of the client computing devices or server computing devices. Furthermore, the network environment 800 can include one or more switches or routers disposed between the other components, such as 880A, 880B, and 880C.

[0116] In addition to the devices described herein, network 830 can include any number of software, hardware, computing, and network components. Additionally, each of the client computing devices, 810, 820, and 830, storage devices 860A and 860B, and server computing devices 870A, 870B, and 870C can in turn include any number of software, hardware, computing, and network components. These components can include, for example, operating systems, applications, network interfaces, input and output interfaces, processors, controllers, memories for storing instructions, memories for storing data, and the like.

[0117] Having described and illustrated the principles of the invention with reference to described embodiments, it will be recognized that the described embodiments can be modified in arrangement and detail without departing from such principles. It should be understood that the aspects of the embodiments described herein are not related or limited to any particular type of computing environment, unless indicated otherwise. Various types of general purpose or specialized computing environments may be used with or perform operations in accordance with the teachings described herein. Elements of the described embodiments shown in software may be implemented in hardware and vice versa, where appropriate and as understood by those skilled in the art.

[0118] As will be appreciated by those of ordinary skilled in the art, the foregoing examples of systems, apparatus and methods may be implemented by suitable program code on a processor-based system, such as general purpose or special purpose computer. It should also be noted that different implementations of the present technique may perform some or all of the steps described herein in different orders or substantially concurrently, that is, in parallel. Furthermore, the func-
tions may be implemented in a variety of programming languages. Such program code, as will be appreciated by those of ordinary skill in the art, may be stored or adapted for storage in one or more non-transitory, tangible machine readable media, such as on memory chips, local or remote hard disks, optical disks or other media, which may be accessed by a processor-based system to execute the stored program code.

[0119] The description herein is presented to enable a person of ordinary skill in the art to make and use the invention. Various modifications to the disclosed embodiments will be readily apparent to those skilled in the art and the generic principles of the disclosed embodiments may be applied to other embodiments, and some features of the disclosed embodiments may be used without the corresponding use of other features. Accordingly, the embodiments described herein should not be limited as disclosed, but should instead be accorded the widest scope consistent with the principles and features described herein.

What is claimed is:

1. An apparatus for accessing protected contents, wherein each of the protected contents is configured to be governed by a license server in a plurality of possible license servers, the apparatus comprising:
   one or more processors; and
   one or more memories operatively coupled to at least one of the one or more processors and having instructions stored thereon that, when executed by at least one of the one or more processors, cause at least one of the one or more processors to:
   receive license location information corresponding to a rights package, the rights package being stored on a license server system and comprising a data structure including content location information and one or more usage rights, wherein the content location information corresponds to a location of an encrypted content package;
   transform the license location information to generate transformed license location information, wherein the transformed license location information includes identity information that identifies at least one aspect of the apparatus;
   determine a license server system indicated by the license location information from a plurality of license server systems;
   transmit the transformed license location information to the license server system, wherein the identity information included in the transformed license location information is configured to be utilized by the license server system to determine whether the apparatus can be trusted to honor a usage license;
   receive a usage license from the license server system including the content location information and the one or more usage rights based at least in part on a determination that the apparatus can be trusted to honor the usage license;
   retrieve the encrypted content package based on the content location information; and
   render content in the encrypted content package in accordance with the one or more usage rights.

2. The apparatus of claim 1, wherein the content location information comprises an address of a content server system and wherein the content location information is received by a sender system from the content server system, packaged with the one or more usage rights to generate the rights package, and transmitted to the license server system.

3. The apparatus of claim 2, wherein the one or more usage rights are determined based at least in part on one or more selections made by a user at the sender system.

4. The apparatus of claim 2, wherein the content location information is received by the sender system in response to a transmission of the encrypted content package from the sender system to the content server system and wherein the encrypted content package is generated at the sender system in response to a selection of content by a user.

5. The apparatus of claim 2, wherein the content location information is packaged with the one or more usage rights by one or more of: the sender system or a distribution server which receives the content location information and the one or more usage rights from the sender system.

6. The apparatus of claim 2, wherein the license location information is transmitted to the sender system from the license server in response to receipt of the rights package.

7. The apparatus of claim 6, wherein the instructions that, when executed by at least one of the one or more processors, cause at least one of the one or more processors to receive license location information corresponding to a rights package further cause at least one of the one or more processors to:
   receive license location information corresponding to the rights package from the sender system.

8. The apparatus of claim 1, wherein the license location information comprises a Uniform Resource Locator (URL) and wherein the instructions that, when executed by at least one of the one or more processors, cause at least one of the one or more processors to transform the license location information to generate transformed license location information further cause at least one of the one or more processors to:
   add one or more parameters identifying the at least one aspect of the apparatus to the URL to generate a transformed URL.

9. The apparatus of claim 8, wherein the at least one aspect comprises information relating to a player application on the apparatus that is configured to render the content.

10. The apparatus of claim 9, wherein the information relating to the player application comprises one or more of: an ecosystem, a version, a user ID, an installation ID, a type of player application, or a content file type.

11. The apparatus of claim 8, wherein the instructions that, when executed by at least one of the one or more processors, cause at least one of the one or more processors to transform the license location information to generate transformed license location information further cause at least one of the one or more processors to:
   add an expiration date to the transformed URL after which the transformed URL can no longer be utilized to retrieve a usage license.

12. The apparatus of claim 11, wherein the instructions that, when executed by at least one of the one or more processors, cause at least one of the one or more processors to transform the license location information to generate transformed license location information further cause at least one of the one or more processors to:
add a signature to the transformed URL, wherein the signature is based at least in part on a portion of the transformed URL.

13. An apparatus for providing a license to protected content, the apparatus comprising:

one or more processors; and

one or more memories operatively coupled to at least one of the one or more processors and having instructions stored thereon that, when executed by at least one of the one or more processors, cause at least one of the one or more processors to:

receive transformed license location information from a client system, the transformed license location information being generated from license location information previously received by the client, wherein the license location information identifies the apparatus and corresponds to a rights package stored on at least one of the one or more memories, wherein the rights package comprises a data structure including content location information and one or more usage rights, wherein the content location information corresponds to a location of an encrypted content package, and wherein the transformed license location information includes identity information that identifies at least one aspect of the client system;

determine whether the client system is trusted to honor a usage license; and

transmit a usage license to the client system including the content location information and the one or more usage rights based at least in part on a determination that the client system is trusted to honor the usage license, wherein the content location information is configured to enable retrieval by the client system of the content in the encrypted content package and wherein the one or more usage rights are configured to govern rendering of the content on the client system.

14. The apparatus of claim 13, wherein the content location information comprises an address of a content server system and wherein the content location information is received by a sender system from the content server system and packaged with the one or more usage rights to generate the rights package, and wherein at least one of the one or more memories has further instructions stored thereon that, when executed by at least one of the one or more processors, cause at least one of the one or more processors to:

receive the rights package.

15. The apparatus of claim 14, wherein the one or more usage rights are determined based at least in part on one or more selections made by a user at the sender system.

16. The apparatus of claim 14, wherein the content location information is received by the sender system in response to a transmission of the encrypted content package from the sender system to the content server system and wherein the encrypted content package is generated at the sender system in response to a selection of content by a user.

17. The apparatus of claim 14, wherein the content location information is packaged with the one or more usage rights by one or more of: the sender system or a distribution server which receives the content location information and the one or more usage rights from the sender system.

18. The apparatus of claim 14, wherein at least one of the one or more memories has further instructions stored thereon that, when executed by at least one of the one or more processors, cause at least one of the one or more processors to:

transmit the license location information to the sender system in response to receipt of the rights package.

19. The apparatus of claim 18, wherein the license location information is received by the client from the sender system.

20. The apparatus of claim 13, wherein the license location information comprises a Uniform Resource Locator (URL) and wherein the transformed license location information is generated by adding one or more parameters identifying the at least one aspect of the apparatus to the URL to generate a transformed URL.

21. The apparatus of claim 20, wherein the at least one aspect comprises information relating to a player application on the apparatus that is configured to render the content.

22. The apparatus of claim 21, wherein the information relating to the player application comprises one or more of: an ecosystem, a version, a user ID, an installation ID, a type of player application, or a content file type.

23. The apparatus of claim 20, wherein the transformed license location information is generated by adding an expiration date to the transformed URL after which the transformed URL can no longer be utilized to retrieve a usage license.

24. The apparatus of claim 23, wherein the transformed license location information is generated by adding a signature to the transformed URL, wherein the signature is based at least in part on a portion of the transformed URL.

25. At least one non-transitory computer-readable medium storing computer-readable instructions that, when executed by one or more computing devices, cause at least one of the one or more computing devices to:

receive license location information corresponding to a rights package, the rights package being stored on a license server system and comprising a data structure including content location information and one or more usage rights, wherein the content location information corresponds to a location of an encrypted content package;

transform the license location information to generate transformed license location information, wherein the transformed license location information includes identity information that identifies at least one aspect of at least one computing device in the one or more computing devices;

determine a license server system indicated by the license location information from a plurality of license server systems;

transmit the transformed license location information to the license server system, wherein the identity information included in the transformed license location information is configured to be utilized by the license server system to determine whether the at least one computing device can be trusted to honor a usage license;

receive a usage license from the license server system including the content location information and the one or more usage rights based at least in part on a determination that the at least one computing device can be trusted to honor the usage license;

retrieve the encrypted content package based on the content location information; and

render content in the encrypted content package in accordance with the one or more usage rights.

26. The at least one non-transitory computer-readable medium of claim 25, wherein the content location information comprises an address of a content server system and
wherein the content location information is received by a sender system from the content server system, packaged with the one or more usage rights to generate the rights package, and transmitted to the license server system.

27. The at least one non-transitory computer-readable medium of claim 26, wherein the one or more usage rights are determined based at least in part on one or more selections made by a user at the sender system.

28. The at least one non-transitory computer-readable medium of claim 26, wherein the content location information is received by the sender system in response to a transmission of the encrypted content package from the sender system to the content server system and wherein the encrypted content package is generated at the sender system in response to a selection of content by a user.

29. The at least one non-transitory computer-readable medium of claim 26, wherein the content location information is packaged with the one or more usage rights by one or more of: the sender system or a distribution server which receives the content location information and the one or more usage rights from the sender system.

30. The at least one non-transitory computer-readable medium of claim 26, wherein the license location information is transmitted to the sender system from the license server in response to receipt of the rights package.

31. The at least one non-transitory computer-readable medium of claim 30, wherein the instructions that, when executed by at least one of the one or more computing devices, cause at least one of the one or more computing devices to receive license location information corresponding to a rights package further cause at least one of the one or more computing devices to:

receive license location information corresponding to the rights package from the sender system.

32. The at least one non-transitory computer-readable medium of claim 25, wherein the license location information comprises a Uniform Resource Locator (URL) and wherein the instructions that, when executed by at least one of the one or more computing devices, cause at least one of the one or more computing devices to transform the license location information to generate transformed license location information further cause at least one of the one or more computing devices to:

add one or more parameters identifying the at least one aspect of the at least one computing device to the URL to generate a transformed URL.

33. The at least one non-transitory computer-readable medium of claim 32, wherein the at least one aspect comprises information relating to a player application on the at least one computing device that is configured to render the content.

34. The at least one non-transitory computer-readable medium of claim 33, wherein the information relating to the player application comprises one or more of: an ecosystem, a version, a user ID, an installation ID, a type of player application, or a content file type.

35. The at least one non-transitory computer-readable medium of claim 32, wherein the instructions that, when executed by at least one of the one or more computing devices, cause at least one of the one or more computing devices to transform the license location information to generate transformed license location information further cause at least one of the one or more computing devices to:

add an expiration date to the transformed URL after which the transformed URL can no longer be utilized to retrieve a usage license.

36. The at least one non-transitory computer-readable medium of claim 35, wherein the instructions that, when executed by at least one of the one or more computing devices, cause at least one of the one or more computing devices to transform the license location information to generate transformed license location information further cause at least one of the one or more computing devices to:

add a signature to the transformed URL, wherein the signature is based at least in part on a portion of the transformed URL.

37. At least one non-transitory computer-readable medium storing computer-readable instructions that, when executed by one or more computing devices, cause at least one of the one or more computing devices to:

receive transformed license location information from a client system, the transformed license location information being generated from license location information previously received by the client, wherein the license location information identifies at least one computing device in the one or more computing devices, and corresponds to a rights package stored on at least one of the one or more memories, wherein the rights package comprises a data structure including content location information and one or more usage rights, wherein the content location information corresponds to a location of an encrypted content package, and wherein the transformed license location information includes identity information that identifies at least one aspect of the client system;

determine whether the client system is trusted to honor a usage license; and

transmit a usage license to the client system including the content location information and the one or more usage rights based at least in part on a determination that the client system is trusted to honor the usage license, wherein the content location information is configured to enable retrieval by the client system of the content in the encrypted content package and wherein the one or more usage rights are configured to govern rendering of the content on the client system.

38. The at least one non-transitory computer-readable medium of claim 37, wherein the content location information comprises an address of a content server system and wherein the content location information is received by a sender system from the content server system and packaged with the one or more usage rights to generate the rights package, and further storing computer-readable instructions that, when executed by at least one of the one or more computing devices, cause at least one of the one or more computing devices to:

receive the rights package.

39. The at least one non-transitory computer-readable medium of claim 38, wherein the one or more usage rights are determined based at least in part on one or more selections made by a user at the sender system.

40. The at least one non-transitory computer-readable medium of claim 38, wherein the content location information is received by the sender system in response to a transmission of the encrypted content package from the sender system to the content server system and wherein the
encrypted content package is generated at the sender system in response to a selection of content by a user.

41. The at least one non-transitory computer-readable medium of claim 38, wherein the content location information is packaged with the one or more usage rights by one or more of: the sender system or a distribution server which receives the content location information and the one or more usage rights from the sender system.

42. The at least one non-transitory computer-readable medium of claim 38, further storing computer-readable instructions that, when executed by at least one of the one or more computing devices, cause at least one of the one or more computing devices to:

transmit the license location information to the sender system in response to receipt of the rights package.

43. The at least one non-transitory computer-readable medium of claim 42, wherein the license location information is received by the client from the sender system.

44. The at least one non-transitory computer-readable medium of claim 37, wherein the license location information comprises a Uniform Resource Locator (URL) and wherein the transformed license location information is generated by adding one or more parameters identifying the at least one aspect of the at least one computing device to the URL to generate a transformed URL.

45. The at least one non-transitory computer-readable medium of claim 44, wherein the at least one aspect comprises information relating to a player application on the at least one computing device that is configured to render the content.

46. The at least one non-transitory computer-readable medium of claim 45, wherein the information relating to the player application comprises one or more of: an ecosystem, a version, a user ID, an installation ID, a type of player application, or a content file type.

47. The at least one non-transitory computer-readable medium of claim 44, wherein the transformed license location information is generated by including in the transformed URL a expiration date corresponding to the transformed URL, wherein the expiration date can no longer be utilized to retrieve a usage license.

48. The at least one non-transitory computer-readable medium of claim 47, wherein the transformed license location information is generated by adding a signature to the transformed URL, wherein the signature is based at least in part on a portion of the transformed URL.

49. A method executed by one or more computing devices for accessing protected contents, the method comprising:

receiving, by at least one of the one or more computing devices, license location information corresponding to a rights package, the rights package being stored on a license server system and comprising a data structure including content location information and one or more usage rights, wherein the content location information corresponds to a location of an encrypted content package;

determining, by at least one of the one or more computing devices, a license server system indicated by the license location information from a plurality of license server systems;

transmitting, by at least one of the one or more computing devices, the transformed license location information to the license server system, wherein the identity information included in the transformed license location information is configured to be utilized by the license server system to determine whether the at least one computing device can be trusted to honor a usage license;

retrieving, by at least one of the one or more computing devices, a usage license from the license server system including the content location information and the one or more usage rights based at least in part on a determination that the at least one computing device can be trusted to honor the usage license;

rendering, by at least one of the one or more computing devices, the encrypted content package based on the content location information; and

rendering, by at least one of the one or more computing devices, content in the encrypted content package in accordance with the one or more usage rights.

50. The method of claim 49, wherein the content location information comprises an address of a content server system wherein the content location information is received by a sender system from the content server system, packaged with the one or more usage rights to generate the rights package, and transmitted to the license server system.

51. The method of claim 50, wherein the one or more usage rights are determined at least in part on one or more selections made by a user at the sender system.

52. The method of claim 50, wherein the content location information is received by the sender system in response to a transmission of the encrypted content package from the sender system to the content server system and wherein the encrypted content package is generated at the sender system in response to a selection of content by a user.

53. The method of claim 50, wherein the content location information is packaged with the one or more usage rights by one or more of: the sender system or a distribution server which receives the content location information and the one or more usage rights from the sender system.

54. The method of claim 50, wherein the license location information is transmitted to the sender system from the license server in response to receipt of the rights package.

55. The method of claim 54, wherein receiving license location information corresponding to a rights package further comprises:

receiving license location information corresponding to the rights package from the sender system.

56. The method of claim 49, wherein the license location information comprises a Uniform Resource Locator (URL) and wherein transforming the license location information to generate the transformed license location information further comprises:

adding one or more parameters identifying the at least one aspect of the at least one computing device to the URL to generate a transformed URL.

57. The method of claim 56, wherein the at least one aspect comprises information relating to a player application on the at least one computing device that is configured to render the content.
58. The method of claim 57, wherein the information relating to the player application comprises one or more of: an ecosystem, a version, a user ID, an installation ID, a type of player application, or a content file type.

59. The method of claim 56, wherein transforming the license location information to generate transformed license location information further comprises:

adding an expiration date to the transformed URL after which the transformed URL can no longer be utilized to retrieve a usage license.

60. The method of claim 59, wherein transforming the license location information to generate transformed license location information further comprises:

adding a signature to the transformed URL, wherein the signature is based at least in part on a portion of the transformed URL.

61. A method executed by one or more computing devices for providing a license to protected content, the method comprising:

receiving, by at least one of the one or more computing devices, transformed license location information from a client system, the transformed license location information being generated from license location information previously received by the client, wherein the license location information identifies at least one computing device in the one or more computing devices, and corresponds to a rights package stored at least one of the one or more memories, wherein the rights package comprises a data structure including content location information and one or more usage rights, wherein the content location information corresponds to a location of an encrypted content package, and wherein the transformed license location information includes identity information that identifies at least one aspect of the client system;

determining, by at least one of the one or more computing devices, whether the client system is trusted to honor a usage license; and

transmitting, by at least one of the one or more computing devices, a usage license to the client system including the content location information and the one or more usage rights based at least in part on a determination that the client system is trusted to honor the usage license, wherein the content location information is configured to enable retrieval by the client system of the content in the encrypted content package and wherein the one or more usage rights are configured to govern rendering of the content on the client system.

62. The method of claim 61, wherein the content location information comprises an address of a content server system and wherein the content location information is received by a sender system from the content server system and packaged with the one or more usage rights to generate the rights package, and further comprising:

receiving, by at least one of the one or more computing devices, the rights package.

63. The method of claim 62, wherein the one or more usage rights are determined based at least in part on one or more selections made by a user at the sender system.

64. The method of claim 62, wherein the content location information is received by the sender system in response to a transmission of the encrypted content package from the sender system to the content server system and wherein the encrypted content package is generated at the sender system in response to a selection of content by a user.

65. The method of claim 62, wherein the content location information is packaged with the one or more usage rights by one or more of: the sender system or a distribution server which receives the content location information and the one or more usage rights from the sender system.

66. The method of claim 62, further comprising:

transmitting, by at least one of the one or more computing devices, the license location information to the sender system in response to receipt of the rights package.

67. The method of claim 66, wherein the license location information is received by the client from the sender system.

68. The method of claim 61, wherein the license location information comprises a Uniform Resource Locator (URL) and wherein the transformed license location information is generated by adding one or more parameters identifying the at least one aspect of the at least one computing device to the URL to generate a transformed URL.

69. The method of claim 68, wherein the at least one aspect comprises information relating to a player application on the at least one computing device that is configured to render the content.

70. The method of claim 69, wherein the information relating to the player application comprises one or more of: an ecosystem, a version, a user ID, an installation ID, a type of player application, or a content file type.

71. The method of claim 68, wherein the transformed license location information is generated by adding an expiration date to the transformed URL after which the transformed URL can no longer be utilized to retrieve a usage license.

72. The method of claim 71, wherein the transformed license location information is generated by adding a signature to the transformed URL, wherein the signature is based at least in part on a portion of the transformed URL.