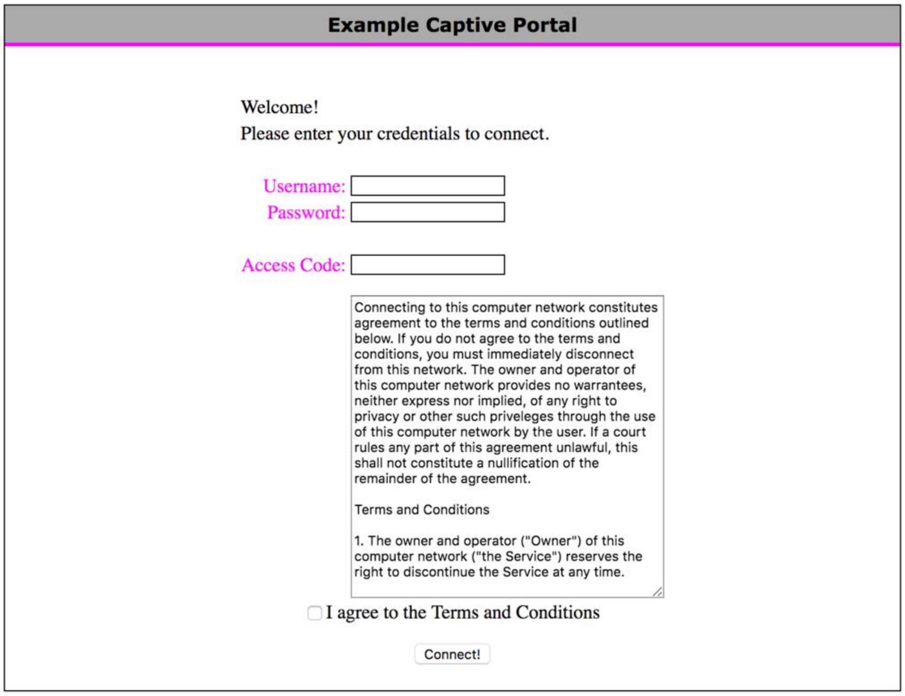
Roman Nicolai

April 10, 2020

ICS 355

Project : Captive Portal (Web Page)

In this report I will be analyzing the Captive Portal web page that is found ubiquitously throughout public Internet settings today. I will be looking at both the technical and legal sides of the Captive Portal. I will analyze how it is designed and implemented from an engineering standpoint as well as research potential legal ramifications around the Captive Portal. Also I will frame the Captive Portal in the techniques introduced in this course. In the context of ICS 355 and Security-Science in general, the Captive Portal is worthwhile to study because of its inherent security capabilities, as well as the fact that it sees great widespread usage in the modern digital world.



Example Captive Portal

Welcome!
Please enter your credentials to connect.

Username:
Password:

Access Code:

Connecting to this computer network constitutes agreement to the terms and conditions outlined below. If you do not agree to the terms and conditions, you must immediately disconnect from this network. The owner and operator of this computer network provides no warrantees, neither express nor implied, of any right to privacy or other such priveleges through the use of this computer network by the user. If a court rules any part of this agreement unlawful, this shall not constitute a nullification of the remainder of the agreement.

Terms and Conditions

1. The owner and operator ("Owner") of this computer network ("the Service") reserves the right to discontinue the Service at any time.

□ I agree to the Terms and Conditions

--Here is an example of a Captive Portal's barebone front-end--

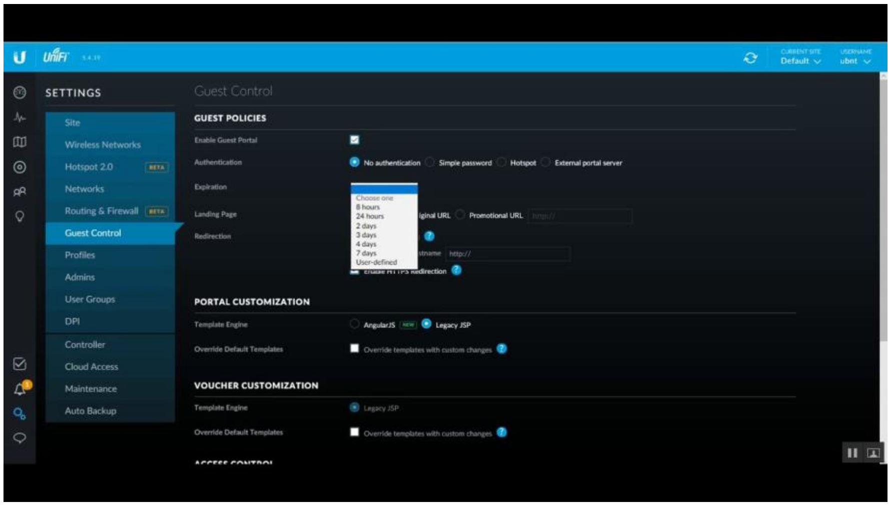
Real World Example: Ubiquiti

To begin with, a Captive Portal is a web page that a user must interact with before gaining complete access to an internet network. In the day-to-day world, Captive Portals are commonly used in public networks with heavy traffic (airports, hotel lobbies, coffee shops, etc). It also sees use in our university. Captive Portals often require a user to authenticate themselves via log-in before allowing access to the network. The Captive Portal may also give other types of information on the page, or even require the user to enter additional data.

Implementing the Captive Portal can range from modest difficulty to requiring high technical experience. It depends on the project and what the user or organization exactly wants or needs. I will go through three examples of Captive Portals ranging in sophistication, these being: UniFi implementation and the implementation used by the University of Hawaii.

--The UniFi AP AC--

UniFi is a wireless networking system from Ubiquiti Networks. There are several products under the UniFi banner that are offered; I will be looking at UniFi AP AC Pro. This device provides a GUI (Graphical User Interface) and an Internet access point. The GUI allows for specific configurations, which includes Captive Portal capability. Once the internet access point is deployed, owners can customize the experience for their users.



Guest Control

GUEST POLICIES

Enable Guest Portal ☑

Authentication ● No authentication ○ Simple password ○ Hotspot ○ External portal server

Expiration: Choose one / 8 hours / 24 hours / 2 days / 3 days / 4 days / 7 days / User-defined

Landing Page iginal URL ○ Promotional URL http://

Redirection

stname http://

Enable HTTPS Redirection

PORTAL CUSTOMIZATION

Template Engine ○ AngularJS NEW ● Legacy JSP

Override Default Templates □ Override templates with custom changes

VOUCHER CUSTOMIZATION

Template Engine ● Legacy JSP

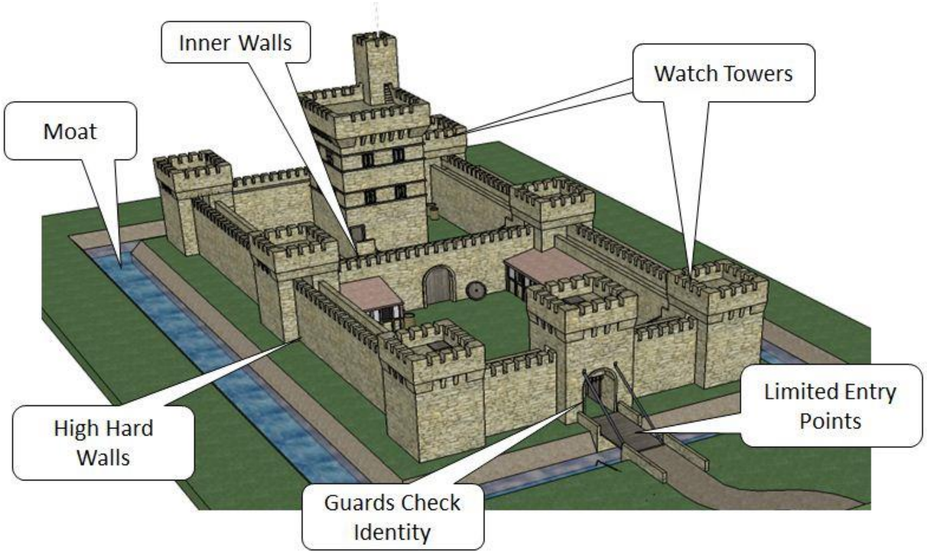
Override Default Templates □ Override templates with custom changes

ACCESS CONTROL

--A customization page within the GUI--

One of the nice features of this Captive Portal is that you can limit certain User Groups' (for example Guests) bandwidth. Bandwidth is how much Internet a user is allowed to use (for example they would not be able to do big downloads if the bandwidth is restricted). Some of the other interesting quirks you can implement include: landing page redirection, expiration timeout, and the portal's visual customization. What makes UniFi AP AC Pro easy to use is its friendly GUI. Also, unlike many other Captive Portal configuration tools, the portal's custom look is not implemented in HTML, rather it is done in an easy GUI feature. Creators can create a quick password to give to users. Once the password is entered correctly, users can gain access to the wireless network. Overall UniFi AP AC Pro is a great entry into Captive Portals for people who do not have much technical experience.

Captive Portal Integration with ICS 355 Material:



--A castle with security measures—

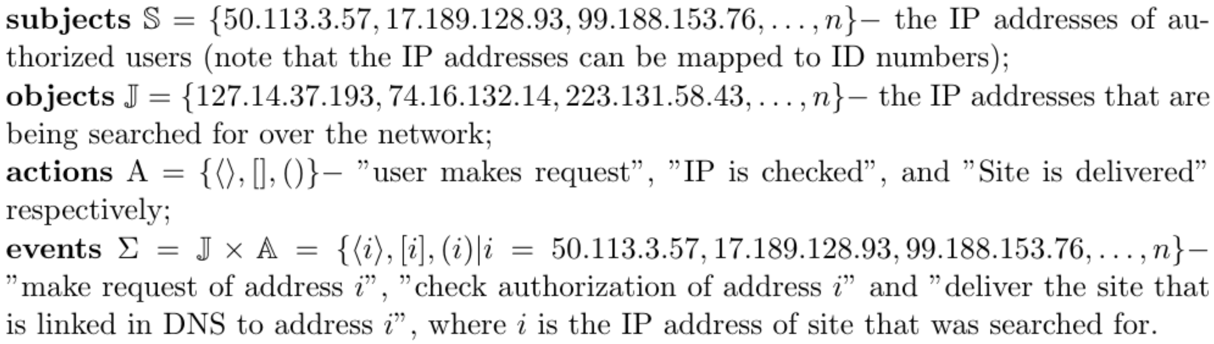
This medieval castle is a symbol for computer security. It is a visual that is useful for applying techniques and concepts shared in ICS 355, and then translating it to a particular real use-case. Throughout the course, I had this castle in my mind while working on assignments, and would model ICS 355 ideas in the context of the castle. For example, the castle subjects could be the authorized commoners, the jesters, the knights, and the royals, all in increasing clearance roles. For this project, the Captive Portal could be the Guards Check Identity and the Limited Entry Points aspects of the castle visual.

I will now model the Captive Portal in the context of the University of Hawaii’s information technology system. UH uses a Captive Portal to authenticate its users before giving access to its Internet network. We can use the concepts learned in ICS 355 to model this system. I will communicate these concepts using the notation of this course as well.

Upon a student registering at UH, they receive a “token” that designates them as a student, which will grant them Internet access. The security perks of this are as follows:

1. The Captive Portal splits users into authorized and non-authorized groups for the University.
2. Once on the network, safety property is satisfied because all users are authorized. All network interactions are safe. Perhaps explicit sites are blocked, which contributes to a safe internet environment for the school. Alternatively, if an explicit search occurs, it can be traced to the student through their log-in credentials. Liveness is satisfied through a strong, healthy network that is well maintained.

Formalization:



subjects $\mathbb{S} = \{50.113.3.57, 17.189.128.93, 99.188.153.76, \ldots, n\}-$ the IP addresses of authorized users (note that the IP addresses can be mapped to ID numbers);
objects $\mathbb{J} = \{127.14.37.193, 74.16.132.14, 223.131.58.43, \ldots, n\}-$ the IP addresses that are being searched for over the network;
actions $\mathrm{A} = \{\langle\rangle, [], ()\}-$ ”user makes request”, ”IP is checked”, and ”Site is delivered” respectively;
events $\Sigma = \mathbb{J} \times \mathbb{A} = \{\langle i\rangle, [i], (i) | i = 50.113.3.57, 17.189.128.93, 99.188.153.76, \ldots, n\}-$ ”make request of address i”, ”check authorization of address i” and ”deliver the site that is linked in DNS to address i”, where i is the IP address of site that was searched for.

Remark!

If the university deems that a safe internet environment means zero explicit searches to sites, then the alternative method would not satisfy the safety property and would not be a viable method. This is because if an explicit search occurs in the history (even if it can be traced to a user’s identity), it would not lead to the safety property (if indeed zero explicit searches constitute safety). Therefore, the method where all explicit sites are blocked is the only viable method.

· **safety**: the site should only be delivered after the authorized user makes a request and the IP is deemed secure.
· **liveness**: the site should eventually be checked and go to the user who requested it.

Remark!

Safety and liveliness would be implemented technically. The Captive Portal ensures the user is authorized, additional measures could be made to ensure that only secure, non-explicit sites are searchable. As for liveliness, a strong network connection all throughout campus would ensure that sites are always delivered to requesting users.

Formalized:

$$\text{SafNet} \ = \left\{\vec{t} \in \Sigma^* | \forall i \in \mathbb{J}.\vec{t} \in \overline{(i)} \implies \vec{t} \in \overline{\exists\langle i\rangle < \exists[i] < (i)}\right\} = \bigcap_{i=0}^{n} \neg\overline{(i)} \cup \overline{\exists\langle i\rangle < \exists[i] < (i)}$$

$$\text{LivNet} \ = \left\{\vec{t} \in \Sigma^* | \forall i \in \mathbb{J}.\vec{t} \in \overline{\langle i\rangle} \implies \vec{t} \in \overline{\langle i\rangle < \exists[i] < \exists(i)}\right\} = \bigcap_{i=0}^{n} \neg\overline{\langle i\rangle} \cup \overline{\langle i\rangle < \exists[i] < \exists(i)}$$

Now we can work with authority and availability properties in the context of the school network. Recall that a process is authorized if bad things do not happen to any of the subjects: *all bad resource requests are rejected* and available if nice things happen for all of the subjects: *all nice resource requests are eventually accepted*.

For our use-case:

· **authorization**: the network should only take some user to a site if they have a clearance and if they requested it.
· **availability**: the network should eventually take every user with a clearance to the site that they requested.

Formalized:

$$\begin{aligned} \text{AuthNet } &= \left\{ \vec{t} \in \Sigma^* | \forall u \in \mathrm{S} \vee i \in \mathrm{J}. \vec{t} \in \overline{(i)_u} \Rightarrow (C\ell(u,i) \wedge \vec{t} \in \overline{\exists \langle i \rangle_u < (i)_u)} \right\} \\ \text{AvailNet } &= \left\{ \vec{t} \in \Sigma^* | \forall u \in S \vee i \in \mathrm{J}. (\vec{t} \in \overline{\langle i \rangle_u} \wedge C\ell(u,i)) \Rightarrow \vec{t} \in \overline{\langle i \rangle_u < \exists (i)_u} \right\} \end{aligned}$$

These localized properties work with our security clearance levels that we will explore in our next section.

3. System administrators and school officials are given higher clearances, which leads to elevated privileges. These higher-ups can access student records, edit the database, and even monitor network activity.

   Formalized:

   security levels $\mathrm{L} = \{\ell_3, \ell_2, \ell_1, \ell_0\}-$ Network Admin/Engineer (ℓ_3), Advisor (ℓ_2), Professor (ℓ_1), Student/Guest (ℓ_0) with $\ell_3 > \ell_2 > \ell_1 > \ell_0$;

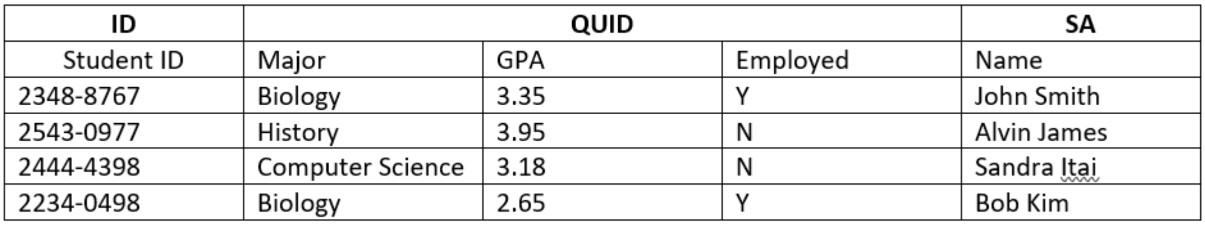
4. The network as a whole is an isolated environment. The activity on the network can be observed as traces and histories. Perhaps the data is cleared once a day at 3AM when the server goes down for maintenance. The traces/histories could be converted to a log form for preservation if disk space is a concern, this creates a chaining of histories:

   $$\vec{z} = \vec{z}_{day1} :: \vec{z}_{day2} :: \cdots :: \vec{z}_{dayN}$$

   We append the current day's history to the previous day, creating a concatenation. System admins can search through the histories if need be. For example, an admin may need to use the system logs to create some type of report, or even catch an explicit user through their actions over the network.

5. How is student data accessed, and kept anonymous? For example, if the school wanted to do research on the student body, but keep student identification unknown, what would be a good

method? A way to do this would be to use *k*-anonymity.

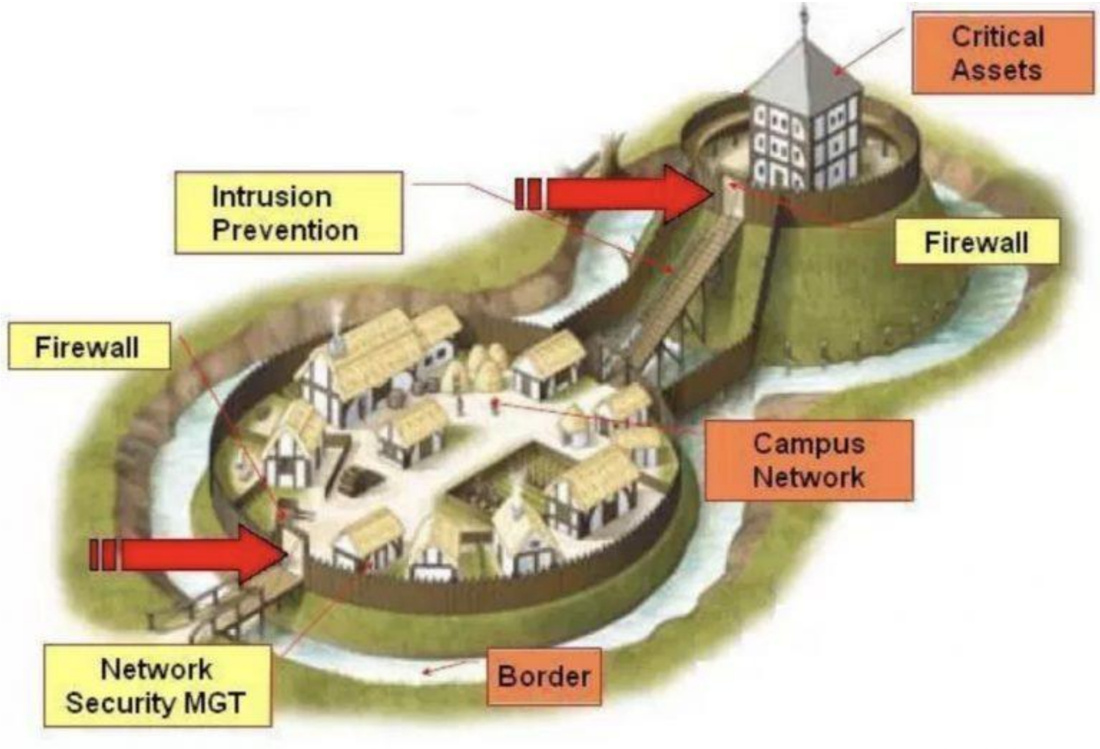


ID	QUID			SA
Student ID	Major	GPA	Employed	Name
2348-8767	Biology	3.35	Y	John Smith
2543-0977	History	3.95	N	Alvin James
2444-4398	Computer Science	3.18	N	Sandra Itai
2234-0498	Biology	2.65	Y	Bob Kim

The quasi-identifier here would be a 3-tuple of {*Major, GPA, Employed-Status*}. The GPA would be entered in a range of .5 increments. An example QUID that could be used on the database would be {Physics, 3.5-4.0, Y}. This would return a group of students belonging to the QUID keys. For example, it could return 5 people belonging to those attributes. Another QUID could be {Biology, 3.0-3.5, N}. This could return 73 students. If the Physics QUID was returned the lowest people in the database, it would give the database a *k*-anonymity rating of 5-anonymity. Because of FERPA (Family Educational Rights and Privacy Act), the student’s identity must never be known during research studies on student population. The sensitive attribute here is the student’s name, and should never be given out. The only time the sensitive attribute should be revealed is in the case of an explicit network violation, where an IP would be traced to a Student ID login, which would then be linked to the student’s name. But this would not be in a research situation. FERPA defends “educational records” such as transcripts, GPA, grades, SSNs, and other information, unless consent is given.

Go to the next page to see further information.

After all this security science notation, our castle now looks like this:



Legality and Legal Analysis of Captive Portals:

Basically, it is legal for private and public entities to set up a wireless local-area, or wireless fidelity, network ("Wi-Fi HotSpot) to provide Internet access to the public or group of users. However, there are numerous legal liabilities that could potentially arise from Captive Portal usage.

A few major ones are as follows:

i. Cyber crimes committed by a third party using the access point.

ii. Content provided to third party users through the wireless access point.

iii. Theft of information from a user because the access is not secure.

iv. Predators harming or luring minors through use of the access point.

To combat these liabilities, there are actions the entity can take. One of these is to block access to certain sites. Another common method is the use of a “splash page”. This splash page can have a use agreement and disclaimer that a user is forced to click through and agree to before gaining use to the network.

The terms and conditions acts as a legal binding contract between the entity and its users. It can prohibit the following and more:

i. Illegal activity such as buying illegal drugs.
ii. Copyright violations such as pirating music and software.
iii. Viewing of pornographic or other explicit content.
iv. Creation of spam, solicitations, advertising.
v. Use of the network by unsupervised minors.

UH has a policy and it is here:

https://www.hawaii.edu/policy/docs/temp/ep2.210.pdf

It is interesting to read through the policy in the context of University usage. For example, the university must comply with acts such as Family Educational Rights and Privacy Act (“FERPA”), since it is dealing with information involving students. Overall, the policy covers a lot of bases and is a good read. Also it is important to point out that users registering for the network do not have to click through or even read the policy to access the network, however they are still bound to following it.

Go to the next page to see Works Cited.

Works Cited

1. Boomtown. “Wi-Fi Hotspots and Liability Concerns.” *Maiello Brungo & Maiello,* Maiello Brungo & Maiello Attorneys at Law, 12 May 2020, www.mbm-law.net/insights/wi-fi-hotspots-and-liability-concerns/.
2. Gebhart, Gennie, and Jacob Hoffman-Andrews. “How Captive Portals Interfere With Wireless Security and Privacy.” *Electronic Frontier Foundation*, 10 Aug. 2017, www.eff.org/deeplinks/2017/08/how-captive-portals-interfere-wireless-security-and-privacy.
3. Pavlovic, Dusko, and Peter-Michael Seidel. “Resource Security: Dynamic Part.” *1. Histories, Properties, Safety and Liveness, 2. Authority and Availability,* 20 Sept. 2019.
4. Pierce, Michelle. “Why Is a Captive Portal Important for Wireless Guest Access?” *WiFi Engineering Services, Software, and Managed WiFi Bundles | SecurEdge WiFi*, SecureEdge Networks, 25 Apr. 2020, www.securedgenetworks.com/blog/why-is-a-captive-portal-important-for-wireless-guest-access.
5. “UniFi - Guest Network, Guest Portal, and Hotspot System.” *Ubiquiti Networks Support and Help Center*, help.ui.com/hc/en-us/articles/115000166827-UniFi-Guest-Network-Guest-Portal-and-Hotspot-System.
6. “What Is a Captive Portal?” *Linksys*, www.linksys.com/us/r/resource-center/captive-portal/.