SEGP2 Plugin Group Milestone 5 Report

Total Hours Spend: approx. 394 hours

Executive Summary

Milestone 5 began by forming an all Masters of Software Engineering students group to focus on developing a functional and usable plugin system for Earth.

The group started with the plan of decomposing the entire development process into three major tasks:

1. Decomposing the File Montior into API methods.
2. Retrofit `earthd` to load plugins from the database.
3. Streamlining the plugins deployment.

Due to time constraints, and the forming of the new group, the group agreed to only complete the first two tasks for this sprint.

At the end of the sprint, `earthd` was retrofitted with the new plugins loading mechanism. Also, the plugin architecture also sport an extension sub-architecture that allows plugins to have plugins as well. All these were integrated in a functional prototype that was demonstrated during the end of milestone presentation.

Besides `earthd`, a few of the members also took the opportunity to explore how the Models, Views Controllers and Helpers can be made pluggable. It was suggested that the entire front-end needs to be revamped to support a toolkit like API architecture. However, this suggestion was dismissed as it is too complicated and there is not enough resources to complete such a feat. The results of the investigations were mixed, and it was agreed that a path will be finalized in the next sprint which the group will work on.

The group achieved synergy by practicing pair, or rather peer, programming. Since Ken has less hours than the rest, the group had gave the administration role to him to run the group, while giving the development responsibilities were assumed by the rest. Besides administrations, Ken was also the main consult for doubts in the plugin system development, as he had been leading this development since Sprint 3.

Overall, the group was able to deliver a solid plugin system for the back-end, and options to be considered for the MVC front-end. Our goal for the next sprint is to finish up the unfinished matters from this sprint and have the daemon plugins system online before the final integration of the entire system.

Individual Reports

Attached are the individual reports of each member of the group.

Ken's Milestone 5 Report

Executive Summary

I have been defaulted as the main contact of the newly formed Plugin Focus Group. Due to my hours being halved due to my commitment on delivering a research paper, I have been assigned to track and manage the group's progress. Besides that, I have initiated a plugin briefing with the group at the start of the group formation, and it was well accepted. The result of the briefing is the current working plugin mechanism, which was demonstrated during the Milestone 5 Meeting, that had exceeded my expectations on how far the new members can produce. On top of that, I had managed to fix a problem that was caused by an unknown bug in the PostgresSQL Ruby connector, and helped move the group forward. Overall, Sprint 5 have been very fruitful and Sprint 6 will be the polishing of what was leftover.

Tasks and Activities Assigned

- Overseeing the progress of the developments of the plugins mechanism development.
  - Sub-task 1: API library composition.
    - Description: Standardize the APIs creation and how to integrate them into plugins.
    - Idea/Solution: N/A
    - Affected files: N/A
    - Git commits: N/A
    - Estimated time taken (planned): (throughout the sprint)
    - Estimated time taken (actual): (throughout the sprint)
  - Sub-task 2: Earth daemon revamping.
    - Description: Retrofitting earthd to support multiple plugins loading from database.
    - Idea/Solution: The main problem of the daemon plugin was due to a translation error between PostgresSQL and Ruby, which caused the pre-signed signatures to corrupt when retrieved as Binary. The solution is to use Text as the storage data type, and pack the codes and signatures into Base64 encoded bundles to maintain consistency. The only thing left to do is to add the decoding process in the Plugin Manager.
    - Affected files: `062_change_plugin_descriptor_code_signature_columns_to_text.rb` and `plugin_manager.rb`
    - Git commits: `(pamalite/earth) ad4cbaed3a0ded49d44453ed674aaaaca6b86144`
    - Estimated time taken (planned): 6 hours
    - Estimated time taken (actual): 7 hours
- Updating the notes briefed during the plugins briefing.
  - Description: Besides having technical briefing about what has been done in previous sprints, on the subject of plugins, some form of a documented notes are needed for easy reference by the new developers. I have prepared some documentations previously, but I saw it is time to update them to reflect the new ideas.
  - Idea/Solution: N/A
  - Affected files: `plugin_developers_guide.pdf` and `Plugin_System_Notes_2.pdf`
  - Git commits: `(pamalite/segp2) 20d0ef6cfc86618a57d08a27078ec8a692921c94`, `(pamalite/segp2) 74b7f3b8c4773935b8038329f5b309417375be1f`
  - Estimated time taken (actual): 3 hours

Resource Contributions

For this milestone, I have estimated that I had spend about 22 hours (including 12 hours meeting time) over the spread of 3 weeks. I have only a budgeted 45 hours (15 hours per week) for this semester. The other 15 hours were spend on my research project.

Looking at the hours I felt that I have had a productive sprint, though I am only using half of my budgeted hours. The rest of the hours were used for other academics commitments. However, I managed to get the new group in-sync with the latest plugins development plans and designs. And, the group managed to exceed the expectation by coming up with a plugin system that works beyond the planned version. Also, the system is readily to be deployed, and it only needs to be polished.

Rooms for Improvement

- Better communication with team members: It was observed that a member of the team had branched off to work on what he feels it is right without the consensus of the group. His action nearly cost the group to repeat history of having too many leads that cannot be followed. Such an action maybe good for the future, but it is bad for the group's reputation when there is no prototype to be shown at the end of the milestone. Besides, having a member who do not share the same vision as everyone else may lead the group into unwanted conflicts.
  - Plan: Level expectations between all group members and myself.

Xiaodong's Milestone 5 Report

Executive Summary

This is the first milestone since we have regrouped. In this new group, I have been working with other MSE students on plugins. I took about one week to understand the existing concepts about plugin, and discussed with others to understand what needed to be done. I also decomposed the filemonitor into some API files based on the API concepts, after all this, I spent some time to look at the GUI plugins as well.

Tasks and Activities Assigned

- Decompose filemonitor into seperated API files
  - Sub-task 1: Looked at the API plugin concepts proposed by other MSE students and the file monitor system.
    - Description: Since I was new to the plugin stuff, it took me some time to look at those stuff, try to understand the whole concept, and had some research on different plugin implementation methods. Also I spent some time to look at the file monitor system in order to find out what can be decomposed to form APIs.
    - Estimated time taken (planned): 25 hours
    - Estimated time taken (actual): 20 hours (19/08/08-24/08/08)
  - Sub-task 2: Decomposed filemonitor into some APIs.
    - Description: We had agreed on the standards of writing API files, and decided that all those API files must be stored under earth_api folder. I decomposed filemonitor into several API files,(Eapi_Eta_printer, Eapi_file_monitor, Eapi_FileMonitor_benchmark, Eapi_logger) and reconstructed filemonitor to use methods defined within those API files.
    - Affected files:
      - `file_monitor.rb`
      - `Eapi_Eta_printer`
      - `Eapi_file_monitor`
      - `Eapi_FileMonitor_benchmark`
      - `Eapi_logger`
    - Estimated time taken (planned): 30 hours
    - Estimated time taken (actual): 22 hours (25/08/08-28/08/08)
  - Sub-task 3: Load filemonitor as a plugin.
    - Description: followed some steps to load filemonitor as a plugin.
    - Estimated time taken (planned): 5 hours
    - Estimated time taken (actual): 3 hours (28/08/08-29/08/08)
- GUI plugin
  - Sub-task 1: Rails plugin could be applied as GUI plugin for earth.
    - Description: I had some research on rails plugin, and I think this can be used to write our GUI plugin.
    - Estimated time taken (planned): 15 hours
    - Estimated time taken (actual): 12 hours (29/08/08-31/08/08)
  - Sub-task 2: Constructed a simple GUI plugin adding a tab into earth, and display some information under the added tab.

- ∗ Description: I created a simple rails plugin (user_usage), included two files one is called user_usage.rb to include a new method (acts_as_add_tab) to allow uses to create a tab, other one is called usage_helper used to display information under user_usage tab.
- ∗ Affected files: user_usage plugin folder
- ∗ Estimated time taken (planned): 6 hour
- ∗ Estimated time taken (actual): 8 hours (31/08/08-02/09/08)

Resource Contributions

For this milestone, I have estimated that I had spend about 71 hours (including 6 hours meeting time) over the spread of 3 weeks. The first week I was almost spent all the time on reading, because I just moved to this plugin group, I needed to understand what they had done for plugins up to milestone 4. The following two weeks I spent the time on decomposing and GUI plugin stuff.

Rooms for Improvement

- I think we need better communication between group members. 4 MSE students were assigned the decomposition part, but we seemed to work on that individually. Some API files created by one person might be created by another one as well. In addition, I worked with GUI plugin, and I believe another MSE student was working on that also, so the resources were split. I think we need to focus on one thing, and use our available resources to make it workable.

Filimoni's Milestone 5 Report

Executive Summary

For the current milestone (M5), I was reassigned to the Plugin Focus Group. This group comprised all the seven (7) MSE students and was tasked with investigating the plugin system for the Earth project. The investigation showed that while the file monitor class (file_monitor.rb) had been designed as a plugin, the actual implementation was tightly coupled to the underlying earth daemon code. This would make it difficult to expand the features and functions of the Earth project. So the investigation also involved modifying the Earth daemon code to make it modular and provide a standardised plugin interface.

Tasks and Activities Assigned

- Examine and Verify Operations of Original Daemon Design and Plugin Support.
  - Sub-task 1: Locate and learn about current group progress with the plugin system.
    - Description: This involves reading the documentation (investigation report) of the plugin system by Group 3 in previous milestones.
    - Idea/Solution: The existing documents revealed that the original code incorporates the option to implement the file monitor class as a plugin.
    - Affected files: `Plugin_Design_Notes.pdf`
    - Git commits: N/A.
    - Estimated time taken (planned): 10 hours
    - Estimated time taken (actual): 16 hours
  - Sub-task 2: Identify all plugin-related designs in codes.
    - Description: While the relevant plugin notes (Sub-task 1) showed the design concepts and explained its implementation, it did not show where these design decisions are implemented in the source codes. As a result, this Sub-task (2) involved identifying the relevant source code files and code snippets in order to attain a better understanding of the original plugin design and implementation.
    - Idea/Solution: N/A.
    - Affected files: `lib/earth_plugins/file_monitor.rb`, `script/earthd`
    - Git commits: `(ssurfer/earth) 0f0464179d5da4d0d22b3e9345bece9aa02136e4`
    - Estimated time taken (planned): 20 hours
    - Estimated time taken (actual): 18 hours
- Eliminate the Tight-Coupling of Earth Daemon and File Monitor Plugin
  - Sub-task 1: Remove hard-coded elements of the file monitor plugin from daemon code.
    - Description: The original form of the Earth daemon involves instantiating the file monitor class and subsequently invoking the (file_monitor) object methods to update the database records of monitored directories.
    - Idea/Solution: This particular Sub-task involved changing this process into a generic approach where the daemon simply invokes a general plugin method (main) that is specified in the EarthPlugin superclass and implemented through method override in all the respective plugin subclasses.
    - Affected files: `lib/earth_plugins/file_monitor.rb`, `script/earthd`
    - Git commits: `(ssurfer/earth) 0f0464179d5da4d0d22b3e9345bece9aa02136e4`
    - Estimated time taken (planned): 30 hours

    - Estimated time taken (actual): 24 hours
  - Sub-task 2: Verify operation of modified daemon and file monitor plugin codes after de-coupling.
    - Description: Verification that de-coupling of the daemon and the file monitor plugin works as planned.
    - Idea/Solution: This Sub-task involved ensuring that in addition to the preservation of the original behaviour of the Earth daemon and file monitor codes implementation, no unnecessary performance degradation occurred as a result of the design modification.
    - Affected files: `lib/earth_plugins/file_monitor.rb`, `script/earthd`
    - Git commits: `(ssurfer/earth) 0f0464179d5da4d0d22b3e9345bece9aa02136e4`
    - Estimated time taken (planned): 10 hour
    - Estimated time taken (actual): 16 hours

- Identify Effective Plugin Design
  - Description: Examined ways to generically implement or load plugins from the daemon.
  - Idea/Solution: The most straight-forward way to implement the plugins was to simply load all existing plugins from the database. Conceptually, this idea also implies that the plugin itself has to have extra-functional features and logic blocks that determines runtime invocation and/or deactivation. The downside of this simplistic implementation means a signficant overhead on the plugin class, which would probably discourage the development of additional feature plugins. A further downside is that any interested plugin developer would need to access and study the daemon code in order to determine how his/her plugin will 'hookup' onto the daemon.
  - Affected files: `lib/earth_plugins/file_monitor.rb`, `script/earthd`
  - Git commits: `(ssurfer/earth) 0f0464179d5da4d0d22b3e9345bece9aa02136e4`
  - Estimated time taken (planned): 20 hours
  - Estimated time take (actual): 12 hours

Resource Contributions

After contributing about 86 hours over the course of 3 weeks during Milestone 5, I believe that more progress could have been achieved if the organisation of existing documents about the plugin system had been up-to-date and less cryptic than what it was at the commencement date of this current development sprint(M5). As a result, a significant portion of this 86 hours was spent on verifying the accuracy and correctness of the documentation.

However, I believe that this situation was only applicable to those of us who were not part of Subgroup 3 during the previous development sprints. Subgroup 1 (my former group) had been exposed to every aspect of the Earth project except the plugin system in previous sprints. This might help explain why a few of us were not as well-versed with the available documentation at the beginning of this sprint. This situation was rectified with the appropriate updates of the plugin system documentation and a special briefing session conducted by Ken.

Room for Improvement

- Better planning: Having coarse-grained Sub-tasks provides room for unproductive investigations that resulted in wasted resources.
  - Plan: Refine subtasks further into 3-4 hours tasks for better monitoring and improved development focus.

<u>Kun Zhou's Milestone 5 Report</u>

Executive Summary

After previous milestone, the main tasks were changed to plugin system and integrated testing. So the MSE students were grouped together to focus on plugin system. Based on the progress of last Gourp3 in milestone 4, we were assigned different task. I and the other 3 students should finish the split-up of existed file monitor and some research on plugin for GUI. The basic API for plugin has been created. But the API system should cost much more time to be perfect in the future. For GUI plugin, I and Jian Huang made a different plugin management system compared with Rails Plugin. The basic function can be used for plug some pages in the Controller side. Therefore, a lot of blocks need to be broken.

Tasks and Activities Assigned

- Daemon API and new File Monitor
  - Sub-task 1: Analyse the File Monitor to find a better way to split it into API
    - This is the most first time for me to analyse the file monitor. I have not touched both of the earthd and file monitor before. That is a big challenge for me to understand the whole system and gain the relationship between different parts of file monitor. Unfortunately, more than 20 hours have been spent on it, including searching information from books and internet.
    - Estimated time taken (planned): 18 hours
    - Estimated time taken (actual): 27 hours
  - Sub-task 2: Create a new file monitor treated as a plugin
    - Description: Honestly, it took much less time than i expected. The whole process is simple. Only two parts of file monitor can be separated into two classes. Obviously, the testing went well.
    - Affected files: `plugin_stuff.tar.gz`
    - Google Group:
      `http://segp2.googlegroups.com/web/plugin_stuff.tar.gz?hl=en&gda=fD840`
      `EUAAAAZYjv4r0sNEon_mhrM-rXNDDviO9ZU7gSmNAhfKCcIXE4EF7-alpc9UU2N5vbkz`
      `bJuoARl3CxVwlYIPn-dXdeGu1iLHeqhw4ZZRj3RjJ_-A`
    - Estimated time taken (planned): 20 hours
    - Estimated time taken (actual): 10 hours

- Run new plugin from database
  - Sub-task1: sign and install the file monitor into database.
    - Description: It is easy for me by following the guider from ken.
    - Estimated time taken (planned): 5 hours
    - Estimated time taken (actual): 5 hours
  - Sub-task 2: Call file monitor from database
    - Description: Unfortunately, this is the biggest problem i have ever meet. The encode problem! I almost spent the whole weekend on this weird problem. The verification is the key for security of plugin in earth system. When the lines of plugin file is less than around 300, the '\n 'would be changed to '\012'. At first I can't understand the reason of the problem. After the discuss in group meeting, ken got a solution using Base64 to encode the file first in order to avoid the appearance of ´
      n: After testing, it is proved as a best solution so fa.

    - Affected files: `plugin_stuff.tar.gz`
    - Google Group: `http://segp2.googlegroups.com/web/plugin_stuff.tar.gz?hl=en&gda=fD840EUAAAAZYjv4r0sNEon_mhrM-rXNDDviO9ZU7gSmNAhfKCcIXE4EF7-alpc9UU2N5vbkzIbJuoARl3CxVwlYIPn-dXdeGu1iLHeqhw4ZZRj3RjJ_-A`
    - Estimated time taken (planned): 18 hours
    - Estimated time taken (actual): 26 hours

- Help Jian Huang to realize his idea on GUI plugin
  - Description: This task was self-assigned.After the meeting with supervisor, I think Jian Huang's idea it possible to solve the plugin problem for GUI. Therefore, I decided to help him and work together for a couple of days on this aim. Now it can be presented but a lot of features are still needed to be done. The created plugin system influence the original earth system huge. So it did not uploaded into the git repository. In the next milestone, the group would discuss the functionality of it and decided whether it can be used for the new GUI plugin system.
  - Estimated time taken (actual): 17 hour

- Low level separation of API
  - Description: This task was self-assigned for make API basically in order to make the new plugin can use it easily. But it haven't started. I would finish it in the next milestone as soon as possible
  - Estimated time taken (actual): 8 hours

Resource Contributions

In this Milestone, I spent too much time on the analysis of file monitor and solution of the strange problem, almost 50 hours. Most of the other hours were mainly spending on working with my group member. But after this milestone, I have a whole structure of earthd including file monitor and understand the way it works.

Rooms for Improvement

The first problem in this milestone for me is my lacking of experience of development about earth daemon. Fortunately, i have passed and got a lot of skills. It should be improved in the future. Additionally, I cost too much time on solution of a particular problem and tried to gain it individually. I should have sent it to the whole group and discusses with the others. It should be a better way to conquer the wall of difficulties.

Mohammad's Milestone 5 Report

Executive Summary

This report records my activities related to Earth project development for Milestone 5 period from 18-August-2008 to 6-September-2008.
In this sprint,I have finished the following main tasks: re-implement the files' metadata feature using RSP tables, design a flexible architecture for (earth daemon) plugins and finally converts file-monitor and my metadata files into plugins for earth daemon.

Tasks and Activities Assigned

- Re-implement the files' metadata feature using RSP tables.
  - Sub-task 1:
    - Description: Adding metadata for files feature was implemented in milestone 4. Now, I re-implemented it using RSP tables. This makes the implementation more efficient and generic.
    - Idea/Solution: avoid redundancy in the database and make all records in the same table having the same type.
    - Affected files:
      lib/earth-plugins/filemonitor.rb,
      lib/earth-plugins/metadata.rb,
      lib/earth-plugins/rsp-metadata.rb and
      lib/earth-plugins/search.rb
    - Git commits: I have created a separate branch in my personal repository for this task named: fileMetadata. There are 12 commits under that branch related to this task:
      You can find all the commits in: http://github.com/mfbDev/earth/commits/fileMetadata.
      They started with
      commits/fileMetadata/0a9ffc836301f9b916. and ended with
      commits/fileMetadata/9063828060000a.
    - Estimated time taken (planned): 18 hours
    - Actual time taken: 20 hours
  - Sub-task 2: testing
    - Description: test the new functionality with the new design
    - Idea/Solution: N/A.
    - Affected files: N/A
    - Git commits: N/A
    - Estimated time taken (planned): 5 hours
    - Actual time taken: 6 hours
- Design a flexible architecture for (earth daemon) plug-ins
  - Sub-task 1: reading
    - Description: read about existing plugin architectures
    - Idea/Solution: N/A.
    - Affected files: extension-points-design.doc
    - Git commits: N/A
    - Estimated time taken (planned): 5 hours

        - Actual time taken: 3 hours
    - Sub-task 2: design and coding
        - Description: design a new plug-in architecture for earth daemon and implement it
        - Idea/Solution: The main idea was using extension-points.
        - Affected files:
          lib/earth-plugin-interface/plugin-manager.rb,
          lib/earth-plugin-interface/extensions.rb,
          lib/earth-plugin-interface/earth-plugin.rb,
          lib/earth-plugins/file-monitor.rb
        - Git commits: There are 10 commits related. All can be found under earth/commits/metadata-as-plugin
          Started with: commits/metadata-as-plugin/9063828060000a
          Ended with: commits/metadata-as-plugin/30d8813d82236c148
        - Estimated time taken (planned): 15 hours
        - Actual time taken: 10 hours
    - Sub-task 3: testing
        - Description: test the new architecture and test file-monitor as a plug-in
        - Estimated time taken (planned): 5 hours
        - Actual time taken: 5 hours

- Add metadata as a plug-in for file-monitor
    - Sub-task 1: adding rsp-metadata as a plug-in for file-monitor
        - Description: create extension points in file-monitor. Link rsp-metadata to these extension points.
        - Affected files:
          lib/earth-plugins/file-monitor.rb,
          lib/earth-plugins/metadata.rb and
          lib/earth-plugins/rsp-metadata.rb,
        - Git commits: There are about 18 commits related.
          All can be found in: earth/commits/metadata-as-plugin.
          Started with:
          commits/metadata-as-plugin/35a48ac6b3f4f92
          Ended with:
          commits/metadata-as-plugin/8434625e07aac85
        - Estimated time taken (planned): 5 hours
        - Actual time taken: 4 hours
    - Sub-task 2: testing
        - Description: testing metadata feature while it is working as a plug-in
        - Affected files: N/A
        - Git commits: N/A
        - Estimated time taken (planned): 2 hours
        - Actual time taken: 4 hours
    - Sub-task 3: loading code from database instead from disk
        - Description: I start the work by loading plug-in files from the disk which make things easier for me. After that, I change the implementation so extensions will load plug-ins from the database.
        - Affected files: same as previous
        - Git commits: included with the previous item
        - Estimated time taken (planned): 1 hours
        - Actual time taken: 3 hours

Resource Contributions

I estimated that I will spent about 70 hours for this milestone. I have spent about 60 hours including meeting times.

Milestone Observations

I believe this milestone was more productive than the previous one because there was more communication between the group members.

Qing Yang Milestone 5 Report

Executive Summary

For this milestone, all the MSE students are regrouped together focusing on the plugin task.I was assigned to the decomposing subgroup. Due to it was my first time to get in touch with the plugin related work,I spent the first week of this sprint in understanding the structure of earth plugin and how earth plugin worked on earth. In the second week ,I thought about how to decomposing the existing plugin filemonitor into apis and then I self-assigned myself to the work of security part of plugin,including creating certificate and keys, sign plugin with the key and put everything into database.However,when I was working on it, I found a bug,but I was not sure whether the bug was from, the ruby::openssl or ruby postgres driver.Then Ken decided to solve the problem by using string to represent the signature rather than the binary.I also spent a lot of time on troubleshooting, The main trouble was that my earth running on Aptenna and earth running in console had a conflict.

Tasks and Activities Assigned

- Create plugin APIs in term of Ken's API standard.
  - Sub-task 1: Have a research on earth plugin.
    - Description:The research includes what is earth plugin , the relationships among filemonitor and other parts of earth, namely ,how they call and work with one another and how plugins work on earth,rather how to create a plugin and how to install a plugin.
    - Affected files:
      `earth/lib/earth_plugins`
      `file_file_monitor.rb`
      `earth/lib/earth_plugin_interface`
      `plugin_manager.rb`
      `earth/script/earthd`
      `earth/script/create_cert`
      `earth/script/sign_plugin`
      `earth/script/install_plugin`
    - Estimated time taken (planned):20 hours
    - Estimated time taken (actual): 21 hours
  - Sub-task 2:Working with group members to decompose the plugin APIs .
    - Description: Discussing with group members and In terms of Ken's API standard, we decomposed the existing plugin file_monitor.rb into APIs.
    - Affected files:
      file_monitor.rb
      api_eta_printer.rb
      api_file_monitor.rb
    - Estimated time taken (planned):10 hours
    - Estimated time taken (actual): 10 hours
- Fixing the conflict between earth running on Aptena and earth running in console.

- Sub-task 1: Find out the reason why I cannot run earth in console by using the command ../script/earthd start
  - Description:I found the socket was occupied by Aptena ,so that earth cannot work on console.
  - Affected files:
  - /tmp/earthd.sock
  - Estimated time taken (planned):6 hours
  - Estimated time taken (actual): 10hours
- Sub-task 2: Uninstall the Aptena Studio plugin on Eclipse
  - Description: At first I did not know how to uninstall it.However, it was done finally.
  - Estimated time taken (planned):5hour
  - Estimated time taken (actual): 7hours

- Work on the security part.
  - Sub-task 1: Learn how to sign and install a plugin.
    - Description: This task included creating certificate,private and public key pair, sign the code of plugin with the private key ,and verify the code and it's signature with the public key in certifucate, when installing a plugin. I spent a lot of time to read the plugin_manager and earthd and learned how to use openssl to do every thing, and felt comfortable to write all the security related functionalities by myself.It indeed took me a lot of time.But When I read Ken's plugin notes,I knew that all I had done were just waste of time.RSP have already finished everything for us.What we need to do was typing create_cert, sign_plugin and install_plugin in command line.
    - Affected files:
      `earth/lib/earth_plugin_interface/plugin_manager.rb`
      `earth/script/earthd`
      `earth/script/create_cert`
      `earth/script/sign_plugin`
      `earth/script/install_plugin`
    - Estimated time taken (planed): 30 hour
    - Estimated time taken (actual): 20 hour
  - Sub-task 2: Put the code,signature, and plugin's name and version into database. And install file monitor as a plugin.
    - Description: Both Keane and I found that if we put the original file_monitor into the database, when we ran ./script/install_plugin the signature can be converted properly which was the same as the code in file_monitor.rb. But once we broke the it into APIs and call the methods that we need from the new created file_monitor file, everything changed. The weired problem is all the '\n' was convered into '\012',so that the code and signature pair could not pass the verification. I spent huge amount of time in testing.I compared the original file_monitor with the new created file_monitor and try to find out the source of the problem. I broke down the original file_monitor little by little,namely the size of file_monitor was becoming smaller and smaller. At first it could work properly, all the '\n' signals were converted in the right way. But when I keeping making its size smaller. The file_monitor was broken by sudden. Therefore, I was sure that this problem depanded on the size of the

file. I considered that it was a bug However,I was not sure where the bug was from, openssl or postgres driver. And finally Ken solved this problem by using string into code and signature in database, instead of binary.

- ∗ Estimated time taken (planned):10hour
- ∗ Estimated time taken (actual): 25hours

Resource Contributions

For this milestone, I have estimated that I had spend about 95 hours (including 4 hours meeting time) over the spread of 3 weeks.

Rooms for Improvement

- We have a big problem in the time and difficulty estimation. And tasks did not assigned to each team member properly.I found that at least half of the team members did not feel comfortable to work in the group.

  - Review more on past milestones, and perform reality checks during planning.

  - Better communication with other members. Team members should trust one another,and be more helpful. Estimation efficiency should be improved.

Huang's Milestone 4 Report

Executive Summary

For this milestone, all the MSE students are regrouped together focusing on the plugin task.I was assigned to the decomposing subgroup. Due to it was my first time to get in touch with the plugin related work,I spent the first week of this sprint in understanding the structure of earth plugin and how earth plugin worked on earth. In the second week ,I thought about how to decomposing the existing plugin filemonitor into apis and then I self-assigned myself to the work of security part of plugin,including creating certificate and keys, sign plugin with the key and put everything into database.However,when I was working on it, I found a bug,but I was not sure whether the bug was from, the ruby::openssl or ruby postgres driver.Then Ken decided to solve the problem by using string to represent the signature rather than the binary.I also spent a lot of time on troubleshooting, The main trouble was that my earth running on Aptenna and earth running in console had a conflict.

Tasks and Activities Assigned

- Create plugin APIs in term of Ken's API standard.
  - Sub-task: Working with group members to decompose the plugin APIs .
    - Description: Discussing with group members and In terms of Ken's API standard, we decomposed the existing plugin file_monitor.rb into APIs.
    - Affected files:
      file_monitor.rb
      api_eta_printer.rb
      api_file_monitor.rb
    - Estimated time taken (planned):5 hours
    - Estimated time taken (actual): 5 hours
- research GUI plugin.
  - Sub-task 1: understand how the rails plugin works
    - Description: read relative books: the rail way, ruby on rail, and so on. After that, briefly illustrate how the rails plugin works, and what's the pro and con, if we use it. Think about my own way to improve it
    - Affected files:
    - /tmp/earthd.sock
    - Estimated time taken (planned):40 hours
    - Estimated time taken (actual): 50hours
  - Sub-task 2: build my way to create Gui plugin instead of rails plugin. In my way, I do not need to change earth coding and Gui plugin can be run in different ways, which will depends on my plugin management.
    - Description: share my idea with group member, discuss it
    - Estimated time taken (planned):20 hours
    - Estimated time taken (actual): 20 hours
- change Ticket 66 which ken has done last semester into a Gui plugin.

  - Sub-task 1: .rebuild ticket 66 as a Gui plugin
    - Description: The reason to choose this ticket is that it involves three parts: view, controller and module. Meanwhile, the most difficult part is change / extend controller, not create a new one.
    - Estimated time taken (planed): 20 hour
    - Estimated time taken (actual): 30 hour
  - Sub-task 2: Test Gui plugin, change some parts of coding again, because what I've got the version of earth has been changed many times, and some parts of coding are not commented.
    - Description: Thank keane for doing my a favour.
    - Estimated time taken (planned):30hour
    - Estimated time taken (actual): 40hours

Resource Contributions

I'm pretty tired in this milestone, because of spending too much time on coding and testing, which means I'm not good at programming.

Rooms for Improvement

- We have a big problem in the time and difficulty estimation. And tasks did not assigned to each team member properly.I found that at least half of the team members did not feel comfortable to work in the group.
  - Review more on past milestones, and perform reality checks during planning.
  - Better communication with other members. Team members should trust one another,and be more helpful. Estimation efficiency should be improved.