|  |  |
| --- | --- |
| **Date:** February 19, 2014 | **Title:** eCash - Vecna Integration Estimate |
| **Author:** Todd Meinershagen | **Contributors:** Kalyan Das, Selva Dinakaran, Mike Faulkinbury, Bao Vu |

1. **Overview**

Currently, eCash is hosted in our internal network. In order to expose the service to external partners such as Vecna, we are proposing to leverage Sentinet, a SOA middleware that allows us to set up a virtual proxy in the DMZ. This will also allow us to take advantage of the A4 security token service for authentication without having to add this to our internal services. Mike Faulkinbury and his team (SOA Infrastructure) will be deploying the DMZ layer of our solution.



1. **Effort**

(As a reference, 8 points = AI Services Team (6 developers/2 qa analysts) \* 2 weeks based on 2014 planning)

| **Feature** | **Description** | **Team** | **Points** |
| --- | --- | --- | --- |
| Modify Interface (Contract) | * Performance – need to do research and/or some load testing * Input - hide a Boolean property on the current contract that indicates whether or not to save credit card info * Input - add the ability to send metadata such as terminal/kiosk * Output – Transaction Id should not be null able; all transactions should have one despite failure or success. * Faults - need to create explicit faults/status codes rather than return default fault with exception details; logging * Regression test eCash service | AI Services | 5 points |
| Deploy –  Partner Integration | * Open firewall ports from Sentinet to eCash * Deploy code to staging environment * Create 14.2 release branch (code freeze) * Create partner documentation * Create partner samples | AI Services | 3 points |
| Configuration – Partner Integration | * Set up mock endpoint for Vecna that returns static response. * Issue certificate in partner integration environment – we will provide the initial certificate and validate * Set up virtual endpoints and security in partner integration Sentinet * Enroll certificate in partner integration – after validation, we will have the client (Vecna) enroll for their own certificate | SOA Infrastructure | N/A |
| Configuration - Production | * Configure production hardware for Sentinet node * Set up DNS for virtual IP * Set up load balancing for 2 node servers * Set up virtual endpoints and security in production Sentinet * Enroll certificate in production – the client (Vecna) will enroll for certificate in production | SOA Infrastructure | N/A |
| Deploy –  Production | * Provision production hardware * Production release activities * Deploy code to production environment * Label 14.2 release branch | AI Services | 2 points |
| **Total Points** |  |  | **10 points** |

1. **Duration**

|  |  |  |
| --- | --- | --- |
| **Sprint** | **# Weeks** | **Features** |
| 1 | 2 | * Modify Interface (Contract) * Deploy – Partner Integration   ***NOTE:*** *Need to request production hardware here.* |
| 2 | 2 | * Configuration – Partner Integration |
|  |  | ***MILESTONE:***   * Contract satisfies customer * Performance satisfies customer * Partner integration validated (with enrolled certificate) |
| 3 | 2 | * Configuration – Production * Deploy - Production |
| **Total:** | **6 weeks** |  |

1. **Hard Costs**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Environment** | **Hardware Needed** | **Shared?** | **Exists?** | **Cost** |
| Partner Integration /  Production | Certificates (2/1) | No | No | $ 0.00 |
| Production | 2 Node Servers - each 2 core, 4GB ram ($6,500 - $8,500/server) | No | No | $ 17,000.00 |
| Production | 2 App Servers – each 2 core, 4GB ram ($6,500 - $8,500/server) | No | No | $17,000.00 |
|  |  |  | **Total Cost:** | **$ 34,000.00** |

1. **Assumptions**

* We are going to create a new version of the ProcessCreditCardPayment() operation in order to preserve backwards compatibility for AccessManager.
* Reporting and billing will work with existing facility information sent for each CC payment because we set up each facility as a unique merchant with Zirmed. If we have the same facility

1. **Risks**

| **Id** | **Description** | **Impact**  (Visibility/Priority) | **Mitigation Plan** |
| --- | --- | --- | --- |
| 1 | May not be able to get the production hardware provisioned in a timely manner based on past experience with IT | 10 | * Have them start building the servers at the beginning of the project so that they have 4 weeks to set it up before needing to validate with client. |
| 2 | eCash may not perform well enough for load (1000/hour) | 7 | * Do some research on average eCash response times in Access Manager to verify if it satisfies Vecna requirements. * Do some performance testing to ensure and do some optimization of code, if needed. * Target Throughput: 1000/hour * **Target Response Time: ??** |
| 3 | Scope may increase after integrating with Vecna. Currently, they only want the Process Credit Card Payment functionality. If they want more later, the timeline will increase. | 5 | * Set up mock partner integration operation that returns static response so that partner can begin testing early to identify if there are issues. |
| 4 | If the same facility will be using Access Manager and Vecna products, when a credit card is processed, we may not be able to differentiate those payments for billing/reporting. (Facility has a direct correlation with Merchant in Zirmed’s configuration) | 2 | * **Ask Jit whether or not this differentiation is important.** * May need to add an Application Id as part of metadata that is tracked for a transaction. |