APPENDIX

Theorem A.1. There is a strict Nash Equilibrium in which, for any computation with a per player reward $\mathit{Reward}_i > \frac{\mathit{cost(calc)}}{\omega - \tau}$, rational computers and requesters follow the protocol.

Proof. Consider a RequestComputation(*requester*, $*$) instance corresponding to a computation instance *calc*, and computers selected for computation I. Based on $n_{comp} > n_{comp}(\psi)$, the majority of computers in I are rational.

First consider a rational requester. Correctly running RevealRewards(*calc*, $*$) allows the requester to run ReturnEscrow(*calc*, $*$) and receive back $\mathit{calc.escrow}_{MM}$. This is because no computer can run a FinaliseRewards(*calc*, $*$) resulting in *claim* $= true$. Therefore, rational requesters follow the protocol

Consider now the rational computers. If the requester correctly runs RevealRewards(*calc*, $*$), *calc.trgt* and $\mathit{calc.responses}_{good}$ are generated correctly. Therefore, if all rational computers follow the protocol, the assumption under which we chose Reward_i in Section V, for a given rational computer $\mathit{computer}_i$ correctly running SubmitResult(*calc*, $*$), $\mathit{computer}_i$ is included in $\mathit{calc.responses}_{good}$ with probability ω. If $\mathit{computer}_i$ incorrectly runs SubmitResult(*calc*, $*$), $\mathit{computer}_i$ is included in $\mathit{calc.responses}_{good}$ with probability of at most τ. By our choice of Reward_i, we have seen in Section V, given $\mathit{calc.responses}_{good}$ is generated correctly and computers included in $\mathit{calc.responses}_{good}$ receive this with probability 1, this is sufficient for rational computers to compute result, equivalent to running SubmitResult(*calc*, $*$).

If the requester correctly runs RevealRewards(*calc*, $*$), $\mathit{calc.responses}_{good}$ is generated correctly. A computer can then run FinaliseRewards(*calc*, $*$) to receive Reward_i if included in $\mathit{calc.responses}_{good}$, as required. If the requester incorrectly runs RevealRewards(*calc*, $*$), any computer can run FinaliseRewards(*calc*, $*$) to generate *claim* $= true$ receive Reward_i with probability 1, which is strictly greater than if the requester correctly runs RevealRewards(*calc*, $*$).

Therefore, rational computers and requesters follow the protocol if $\mathit{Reward}_i > \frac{\mathit{cost(calc)}}{\omega - \tau}$ ⊠

Lemma A.2. For a series of computations $[\mathit{calc}_1, \mathit{calc}_2, ..., \mathit{calc}_n]$ with $\mathit{Reward}_i > \frac{\mathit{cost(calc)}}{\omega - \tau}$ and $n_{comp} > n_{comp}(\psi)$, as the number of completed computations increases, the probability of selecting a Byzantine computer for a computation with $n_{comp} < \frac{\mathit{Computers}}{2}$ is strictly decreasing in expectancy and approaches 0 as n tends to infinity.

Proof. As $\mathit{Reward}_i > \frac{\mathit{cost(calc)}}{\omega - \tau}$, from Theorem A.1 rational computers follow the protocol. Let α be the share of computers that are Byzantine. We know a majority of computers selected are rational, as $n_{comp} > n_{comp}(\psi)$. Therefore, Byzantine computers are rewarded with probability $\tau < \omega$. For a given computation, the expected reputation increase of a selected Byzantine computer is τ, while the expected increase for a selected rational computer is ω. Given n_{comp} are selected for the computation, the expected number of these being rational computers is $(1 - \alpha)n_{comp}$, while the number of selected Byzantine computers is αn_{comp}. Furthermore, this means the expected increase in reputation for rational computers is $(1 - \alpha)n_{comp}\omega$, while the expected increase in reputation for Byzantine computers is $\alpha n_{comp}\tau$. At the beginning of the protocol, the probability of selecting a Byzantine player from the set of all computers is in direct proportion to starting reputation. Given initial reputations of *initRep*, after the first computation, the selection probability of a Byzantine computer reduces in expectancy to:

$$\frac{\alpha(|\mathit{Computers}|\mathit{initRep} + n_{comp}\tau)}{|\mathit{Computers}|\mathit{initRep} + n_{comp}\big((1-\alpha)\omega + \alpha\tau\big)}. \qquad (5)$$

First it be can see that

$$\frac{\alpha(|\mathit{Computers}|\mathit{initRep} + n_{comp}\tau)}{|\mathit{Computers}|\mathit{initRep} + n_{comp}\big((1-\alpha)\omega + \alpha\tau\big)} < \alpha \qquad (6)$$

meaning Byzantine selection probability is decreasing. To prove that Byzantine selection probability tends to 0 in the number of computations as described in the Lemma statements, let α_k be the Byzantine computer selection probability after k computations. We have the expected Byzantine selection probability after $k+1$ computations, denoted , α_{k+1}, is:

$$\frac{\alpha_k(|\mathit{Computers}|\mathit{initRep} + n_{comp}\tau)}{|\mathit{Computers}|\mathit{initRep} + n_{comp}\big((1-\alpha_k)\omega + \alpha_k\tau\big)} = \frac{\alpha_k(|\mathit{Computers}|\mathit{initRep} + \tau n_{comp})}{|\mathit{Computers}|\mathit{initRep} + n_{comp}\omega - \alpha_k n_{comp}(\omega - \tau)}. \qquad (7)$$

We have already seen:

$$\alpha_{k+1} = \frac{\alpha_k(|\mathit{Computers}|\mathit{initRep} + n_{comp}\tau)}{|\mathit{Computers}|\mathit{initRep} + n_{comp}\omega - \alpha_k n_{comp}(\omega - \tau)} < \alpha_k. \qquad (8)$$

which implies:

$$\frac{(|\mathit{Computers}|\mathit{initRep} + n_{comp}\tau)}{|\mathit{Computers}|\mathit{initRep} + n_{comp}\omega - \alpha_k n_{comp}(\omega - \tau)} < 1. \qquad (9)$$

Letting the term on the right be r_k, we can see r_k is decreasing in k as $n_{comp}(\omega - \tau) > 0$ (because $\omega > \tau$) and $\alpha_{k+1} < \alpha_k$, meaning the negative term in the denominator of r_k is increasing (towards 0) and as such the denominator of r_k is increasing. Therefore $\alpha_k < \alpha_0 r_0^k$, with $r_0 < 1$. The result follows. ⊠