We thank the reviewers for taking their time to carefully review our paper, for considering our attack method to be novel and interesting, and for suggesting improvements.

Our main point is indeed showing that existing robust-statistics methods are not enough in many real-world use cases. We find significance in identifying assumptions that do not hold in the real world. Rev2 described it well in their review.

As a method to foretell the success rate of the attack, the attacker can check the ratio of parameters for which the gradients can change direction with the given attack. This is applicable for a dimension j when the size of the mean gradient ($|\mu_j|$) is smaller than the change that the attacker can introduce: $z\sigma_j$, for the z described in the paper and σ_j the standard deviation of the gradients across the different workers. Figure 1 shows the calculations on the 3 experimented tasks with $z \in \{\frac{1}{2}, 1, 2\}$. It is clear that many gradients can change direction even with a change of only $\frac{1}{2}\sigma$. These results negate the assumption of most defenses (most explicitly by Krum) that the standard deviation is smaller than the gradient itself. We will extend this analysis including discussion on backdooring for the final version.

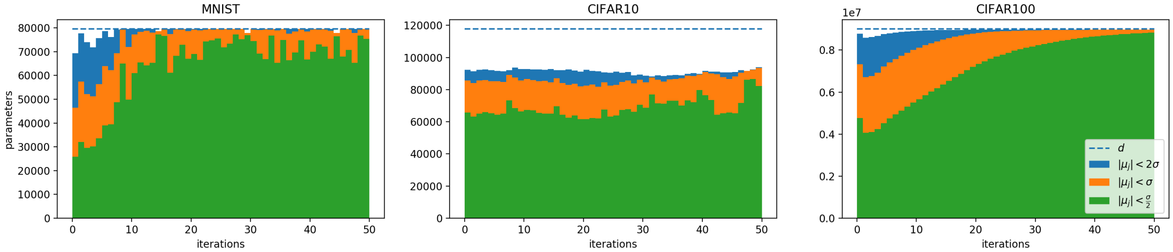

Figure 1: Number of parameters per iteration where $|\mu_j| < z\sigma_j$, for $z \in \{\frac{1}{2}, 1, 2\}$ in the first 50 iterations.

We would like to thank Rev2 for bringing DRACO to our attention. We were not familiar with that work, which we surely should and will discuss for the camera-ready version, as it should be resilient to our attack. DRACO takes a coding-based approach instead of the robust-aggregation approach used by the defenses referred to in our paper. It is achieved by having the PS sending each chunk of data to multiple workers, and using majority to find the correct evaluation of each chunk. In that regards we have 2 remarks:

a) DRACO defends against a very limited number of byzantine workers, not $\mathcal{O}(n)$ such as Krum, TrimmedMean or Bulyan. For our experiments where $m = 12$ workers were corrupted, each chunk needs to be calculated $r = m*2+1 = 25$ times. This implies that training process that should have taken 2 days without defense will take almost 2 months with DRACO. We disagree with the authors of this defense that only a few workers can be corrupted in real life. For example, an attacker controlling a network component (e.g. a router or a switch) near the PS will be able to perform a Man-In-The-Middle attack by adopting our method while controlling more than a handful of nodes.

b) DRACO does not prove its superiority over robust-statistics methods in the face of a specific attack. We show that DRACO's contribution is more significant than run-time improvement, because only methods that force the results to be **identical** to results without Byzantine workers such as DRACO will be resilient to attacks that require minimal changes. Consecutively, we will limit our claims for overcoming existing defenses based on robust-statistics methods only.

Models and datasets chosen: First, for MNIST and CIFAR10 we followed the models and hyper-parameters selected by the authors of Bulyan (the baseline defense we overcome) for a fair comparison. We added CIFAR100 with WideResNet architecture in order to test our attack on a more realistic task. Our results show that our attack works on all ranges of tasks, from the simplest (MNIST with 2 fully connected layers), through quite simple ConvNet for CIFAR10, to the much more complex WideResNet architecture on CIFAR100 dataset. We find it interesting to present results on MNIST, showing that the variances are high enough even for such a simple task. As of the request for more experiments with different settings, we would like to point the reviewers to the supplementary material, where we evaluated our methods on different number of Byzantine nodes (Figure 2) and different shifting factors (Table 1). We can also add results which were not included initially showing that the results are similar for different number of overall workers (n).

Non-omniscience: All m corrupted workers report **only their private data** to the attacker, meaning that the attacker is indeed non-omniscient. We will reiterate it for the final version.

Real-World Relevance: Deep Learning became a research field which involves tremendous amount of money, and various players are motivated to prevent a company from reaching high accuracy on some task, or backdooring their model. This includes hacking into many training nodes or a central networking component as mentioned above. Another trend amplifying this risk is the rise of start-ups allowing people to get paid for hosting training tasks on their private GPUs in a distributed fashion while idle. In such model, the workers obviously cannot be trusted.

We will revise all cases which are pointed by the reviewers to be misleading, odd or unclear. We will also fix typos.